Self-adaptive *hp* finite element method with iterative mesh truncation technique accelerated with Adaptive Cross Approximation

R.M. Barrio-Garrido , L.E. Garcia-Castillo , I. Gomez-Revuelto , M. Salazar-Palma

Keywords:
Finite element method
Adaptive Cross Approximation
hp-adaptivity
Open region scattering problems
Fast method

A B S T R A C T

To alleviate the computational bottleneck of a powerful two-dimensional self-adaptive *hp* finite element method (FEM) for the analysis of open region problems, which uses an iterative computation of the Integral Equation over a fictitious boundary for truncating the FEM domain, we propose the use of Adaptive Cross Approximation (ACA) to effectively accelerate the computation of the Integral Equation. It will be shown that in this context ACA exhibits a robust behavior, yields good accuracy and compression levels up to 90%, and provides a good fair control of the approximants, which is a crucial advantage for *hp* adaptivity. Theoretical and empirical results of performance (computational complexity) comparing the accelerated and non-accelerated versions of the method are presented. Several canonical scenarios are addressed to resemble the behavior of ACA with *h*, *p* and *hp* adaptive strategies, and higher order methods in general.

1. Introduction

Self-adaptive *hp* finite elements are a powerful tool to analyze a wide variety of problems in physics and engineering, [1,2]. From an initial mesh, and in a fully automatic fashion, *h*-refinements (modification of element size) and *p*-refinements (variation of the polynomial order *p*) are performed simultaneously, providing exponential rates of convergence even in the presence of singularities. In contrast, only algebraic rates of convergence are in general obtained with *h* or *p* adaptive schemes. With respect to *hp*-adaptivity for electromagnetic problems, extra technicalities come into play due to the particular characteristics of Maxwell's equations: development of hierarchical curl-conforming elements, commutativity of the *de Rham* diagram, and so on (a nice condensed summary can be found in [3]).

On the other hand, when dealing with open region wave problems, the FEM requires to artificially truncate the infinite original domain in order to keep the number of unknowns finite. In this context, the authors have developed an iterative methodology for the FEM analysis of open region wave propagation problems with easy hybridization with other methods [4–6]. This methodology is referred here to as Finite Element-Iterative Integral Equation Evaluation (FE-IIEE). It

makes use of an Integral Equation (IE) representation of the electromagnetic field in the exterior region (out of the truncated domain). However, in contrast to conventional FEM-IE approaches, this is done in such a way that the original sparse structure of the FEM matrices is retained. FE-IIEE has demonstrated to be suitable for *hp*-adaptivity [7], as it provides an adjustable, and arbitrarily exact, radiation boundary condition for the wave propagation problem. For further reading the interested reader is referred to [8] in which a comparison study between FE-IIEE and other truncation techniques suitable for *hp*-adaptive methods such as the use of infinite elements [9] or the use of Perfect matched Layers (PML) [10], is presented. FE-IIEE key advantages are achieved at the expense of performing a few iterations in which the exterior field is calculated using the IE expressions and the Green's function of the exterior problem. As it will be clear later, the convolutional character (double loop) of these computations leads to a computational complexity of $O(N^2)$, being N the number of unknowns associated to the exterior boundary.

Several methods have appeared in the literature to accelerate these type of integral computations, such as the Fast Multipole Method (FMM) [11,12], grid approaches based on Fast Fourier Transform (FFT) (e.g., [13–16]), and those based on algebraic compression (e.g., based on QR factorization [17], Adaptive Cross Approximation (ACA), [18,19], and, in general, algorithms related to *hierarchical matrices*, [20,21]).

Methods based on FFT over regular grids are not the best suited for *hp*-adaptivity due precisely to the irregularities of the *hp* adapted meshes (careful implementations of projections between the *hp*-FEM grid and the regular grid of the FFT should be coded). On the other hand, multipoles and algebraic methods do not suffer from this drawback.

FMM is a powerful method that has demonstrated to reduce the computational cost from $O(N^2)$ to $O(N^{1.5})$ for a single level implementation and to $O(N \log N)$ in a multilevel implementation, [22–24]. However, FMM is problem dependent and requires a very careful choice of the error control parameters, which can only be partially adjusted. The latter feature is a serious drawback in the context of *hp*-adaptivity in general and, specifically, with an *hp* version of FE-IIEE in which two iterative loops are nested. Pure algebraic methods are based exclusively on the rank deficient structure of the matrix obtained from the discretization of the IE operator; thus, they are problem independent. Specifically, ACA provides a rank revealing algorithm (on the fly) that generates blockwise low-rank approximants from the boundary IE matrices. ACA is robust regarding the clusterization scheme and other physical underlying factors, and also provides a robust and straightforward error control of the approximants, giving a more flexible control of the accuracy than FMM, along with an easier implementation. This is a crucial advantage when combining with automatic *hp*-adaptivity.

In this paper, the self-adaptive *hp* FE-IIEE presented in [7] is accelerated with the ACA algorithm. A specific implementation of ACA suited to the use of *hp*-grids has been developed. Results of the effect of the ACA approximation on the final error delivered by the method and the interaction with each one of the two iterative loops (*hp* and FE-IIEE) present in the method are shown. The main contribution of the paper is to show the viability and main features of the implementation of the ACA algorithm in the *hp* context. Detailed empirical results in terms of computational complexity under a number of representative scenarios of *h*, *p* and *hp* adaptive strategies, and higher order methods in general, are shown. The method is presented and tested on a two dimensional (2D) layout; being the conclusions of this work easily extrapolated to the general approach in the three-dimensional (3D) case. The reason for the lack of 3D results is mainly that the 3D implementation of *hp* adaptivity for electromagnetic problems is under intensive development and the most stable recent version [25,26] does not include FE-IIEE yet.

2. Methodology

The automatic *hp*-adaptive strategy for open region problems is based on the algorithm presented in [7], which will be briefly reviewed here. Details of the formulation will be given later in Section 3.

FE-IIEE is used as the mesh truncation methodology. FE-IIEE is based on a two domain decomposition multiplicative Schwarz paradigm [27]. The original infinite domain is divided into two overlapping domains: a finite FEM domain (Ω^{FEM}) bounded by surface S and the infinite domain exterior to the auxiliary boundary S' (Ω^{EXT}). Thus, the overlapping region is limited by S' and S. An example of FE-IIEE domain decomposition setup is shown in Fig. 1. It corresponds to the analysis of the scattering of electromagnetic waves on infinitely long z-oriented cylinders of arbitrary geometry and material configuration. This is the problem that we have chosen in this paper to illustrate the validity and performance of ACA acceleration on the self-adaptive *hp*-FEM. The incident wave is assumed to be propagating along a direction orthogonal to z; furthermore, z-variation of the excitation is assumed to be null. Plane waves and cylindrical waves propagating orthogonal to z-axis are examples of such incident waves. In this case, and in contrast to a general 3D problem with translational symmetry (e.g., inhomogeneous waveguides [28]) in which the problem needs to be formulated as a coupled system of equations on the 2D cross section of the 3D problem, the problem at hand can be modeled with just two uncoupled scalar Helmholtz equations in which the unknown is the longitudinal (z) component of the field; specifically, the electrical field for the so called Transverse Magnetic (TM) polarization and the magnetic field for the so called Transverse Electric (TE) polarization [29, chapter 11]. Any other case can be expressed as a linear combination of TM and TE polarizations. It is worth noting that the same TM and TE scalar equations can also be used to model the radiation of electromagnetic waves originating from impressed longitudinal currents with arbitrary distribution on the transverse components (x, y) but invariant with z. This is also depicted in Fig. 1.

Following Schwarz paradigm, FE-IIEE iterates between FEM domain and exterior domain solutions until the error between the field ϕ provided by both domain models in the overlapping region is below a certain threshold U. On the

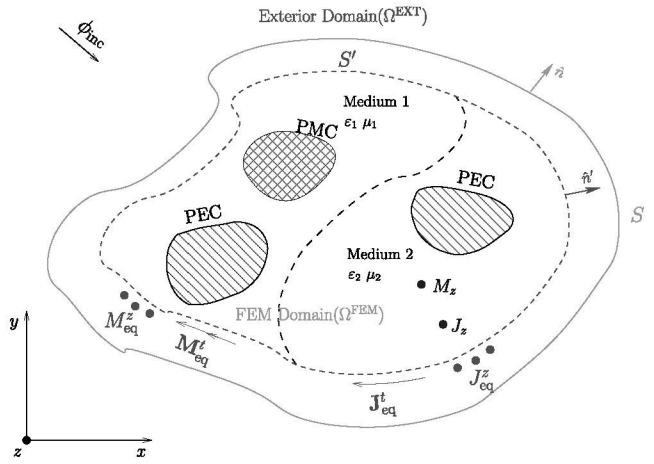

Fig. 1. Cross section of scattering/radiation problem with translational symmetry showing the domain decomposition setup of the resulting 2D open region problem.

other hand, the automatic *hp* adaptivity is an iterative process too. Thus, the inclusion of FE-IIEE in the self-adaptive *hp*-FEM yields a doubly nested loop algorithm (see Fig. 2).

The self-adaptive strategy iterates along the following steps. First, a given *coarse hp*-mesh is globally refined both in h and p to yield a *fine* mesh. Then, the problem of interest is solved on the coarse and fine meshes. ϕ_C and ϕ_F are used to denote the field solution on the coarse and fine meshes, respectively. The difference between the fine and coarse grid solutions provides an error function (an error indicator is not enough) that is used to guide optimal refinements over the coarse grid (block "New Optimal Coarse Mesh" in Fig. 2). Details are quite involved and can be found in [1]. Roughly speaking, a "competition" between p-refinement with all *competitive* h-refinements takes place at each iteration step. The competitive h-refinements are those that result in the same increase in the number of degrees of freedom (d.o.f.) as the p-refinement. The "competition" is driven by the *error decrease rate* (EDR) of each edge and element of the mesh, which interpolates the error from the fine mesh to the coarse one. This operator is local, it maintains conformity and it is optimal in the sense that the error behaves asymptotically, both in h and p, in the same way as the actual interpolation error (see [1,30] for details). The process finishes when the error of the coarse mesh with respect to the fine mesh, e_{hp}, is lower than the user defined T parameter, which establishes the minimum accuracy to be achieved. Note that the green dashed line box at the top of the flow chart of Fig. 2 includes the algorithm corresponding to the automatic *hp*-adaptivity.

Thus, the outer loop provides an optimal FEM *hp*-discretization for a given "continuous" problem cast in variational form. The FE-IIEE inner loop takes action in the solve phases of both, the coarse and the fine mesh, solutions. As it will be clear later in Section 3, FE-IIEE iteratively updates the right hand side (RHS) of the *hp* discrete problem providing an arbitrarily accurate mesh truncation boundary condition for scattering/radiation problems. The relevant part of the flow chart of FE-IIEE is shown as a module (dashed box at the bottom). The use of FE-IIEE in the solve phase corresponding to the fine mesh does not imply a restart of the FE-IIEE iterations. A projection of the RHS of the coarse mesh onto the fine mesh is coded providing an efficient implementation of FE-IIEE within the *hp*-automatic strategy. For further details the reader is referred to [7,31].

In terms of the errors involved, the behavior of the doubly nested loop is the following. Although the inner loop works at the discrete level, the outer loop of FE-IIEE behaves as the one providing an exact boundary condition at the continuous level to the *hp*-FEM discretization. This is true provided that the error of the inner loop is low enough with respect to the error of the outer loop, i.e., $U < e_{hp}$, where for the final mesh $e_{hp} < T$ (see Fig. 2). It has been shown that, typically, a ratio of one order of magnitude between T and U is more than enough (see [7]).

The ACA compression algorithm provides an accelerated method for calculating the contribution to the RHS of the discrete problem corresponding to the radiation boundary condition. Furthermore, the error introduced by ACA is easily controllable with only one parameter (referred to as ϵ). Thus, the value of ϵ can be set in an adaptive fashion as a function of the present threshold values U and T. Examples of the viability of ACA as fast method in *hp*-adaptive scenarios are shown in Section 5.1.

3. Formulation

As it was previously mentioned in Section 2, the analysis of the scattering/radiation of electromagnetic waves on infinitely long z-oriented cylinders of arbitrary geometry and material configuration can be modeled with a 2D scalar Helmholtz equation in terms of the longitudinal component of the field. That is true provided the excitation is invariant with z. Depending on which of the two canonical cases are considered (TM or TE), the unknown field is the electric field or the magnetic field, respectively (see, for instance, [32, chapter 3], [33, chapter 4]). Thus, FE-IIEE setup for this problem

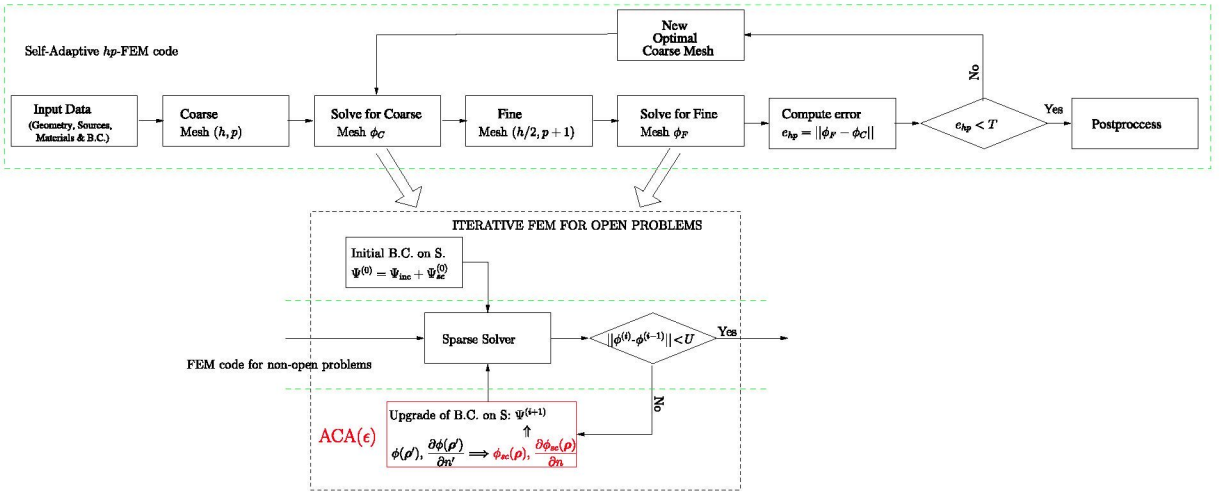

Fig. 2. Flow chart of the self-adaptive *hp* methodology combined with FE-IIEE and ACA. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 1
Correspondences for scattering and radiation formulations.

Pol.	ϕ	f_r	g_r	I_o	q
TM	E_z	μ_r	ϵ_r	η_o	$jk_0 I_0 J_z$
TE	H_z	ϵ_r	μ_r	$1/\eta_o$	$jk_0 I_0 M_z$
Pol.	L^t_{eq}		O^z_{eq}		
TM	$\mathbf{M}^t_{\text{eq}} = \phi$		$J^z_{\text{eq}} = \left(\partial\phi/\partial n'\right)/(jk_0 I_0)$		
TE	$\mathbf{J}^t_{\text{eq}} = \phi$		$M^z_{\text{eq}} = \left(\partial\phi/\partial n'\right)/(jk_0 I_0)$		

corresponds to the one shown in Fig. 1. The field in Ω^{FEM} is modeled in the frequency domain by Helmholtz equation that can be expressed for both, TM and TE, polarizations as,

$$\nabla \cdot \left[f_r^{-1} \nabla \phi\right] + k_0^2 g_r \phi = q \tag{1}$$

where k_0 is the vacuum wavenumber which is the square function of the angular frequency and the permittivity and permeability of vacuum. The correspondences for the remaining magnitudes and parameters are given in Table 1: E, H denote electric and magnetic field, respectively, and ϵ_r, μ_r electric permittivity and magnetic permeability, respectively, relative to vacuum medium. Symbols J and M denote electrical and magnetic currents, respectively. Symbol η_0 is used to denote the intrinsic impedance of vacuum medium which is equal to 120π. Symbol q represents internal sources (radiation problem). The exterior excitation (scattering problem) is achieved by the exterior boundary condition as it is explained next.

The following conditions (B.C.) are used:

$$\phi\left(\boldsymbol{\rho}\right) = 0 \quad \boldsymbol{\rho} \in \Gamma_D \tag{2}$$

$$\frac{\partial \phi\left(\boldsymbol{\rho}\right)}{\partial n} = 0 \quad \boldsymbol{\rho} \in \Gamma_N \tag{3}$$

$$\frac{\partial \phi(\boldsymbol{\rho})}{\partial n} + j\, k_0\, \phi(\boldsymbol{\rho}) = \Psi\left(\boldsymbol{\rho}\right) \quad \boldsymbol{\rho} \in \Gamma_S \tag{4}$$

where symbol n stands for the outward normal to the corresponding boundary, $\boldsymbol{\rho}$ is the position vector in cylindrical coordinates and Ψ is a function. Symbol Γ refers to Ω^{FEM} boundary; specifically, Γ_D refers to Dirichlet boundaries, Γ_N refers to Neumann boundaries and Γ_S refers to boundary S.

Dirichlet (2) and Neumann (3) B.C. are used to model Perfect Electric Conductors (PEC) and Perfect Magnetic Conductors (PMC). With the electric field formulation (TM), PEC contours are B.C. of Dirichlet type while PMC contours are B.C. of Neumann type. With the magnetic field formulation (TE), is the other way around by duality. PMCs do not exist in nature but they are used in numerical modeling, e.g., as symmetry walls.

Note that a local B.C. of the Cauchy type on the outer limit of the FEM domain, S, is used. The residual function $\Psi(\boldsymbol{\rho})$ is estimated by FE-IIEE as it will be clear later. The use of the Cauchy B.C. avoids the interior resonance problem [32, chapter 6] and improves the convergence of the FE-IIEE method.

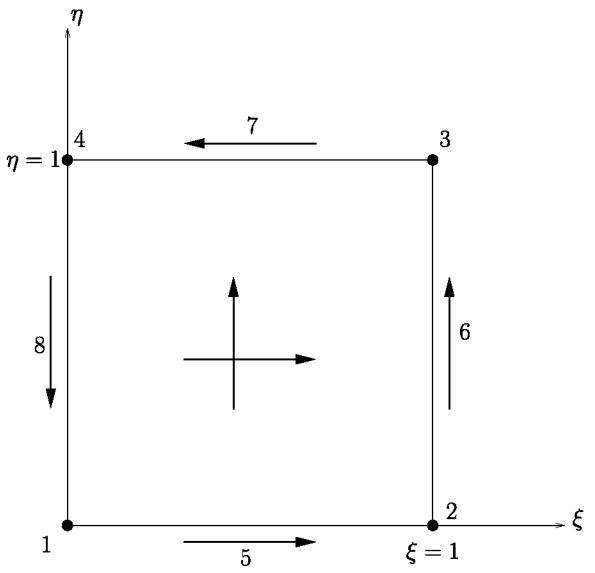

Fig. 3. Quadrilateral master finite element.

The variational formulation of the problem is then obtained by multiplying (1) by a suitable test function ω, using integration by parts, and finally including boundary conditions in the resulting integral formulation. Thus, the formulation of the problem reads:
Find $\phi \in H^1$ such that

$$b(\omega, \phi) = f(\omega) \quad \forall \omega \in H_0^1 \tag{5}$$

where $H_0^1 := \{p \in H^1(\Omega),\ p = 0 \text{ on } \Gamma_D\}$, bilinear form b is defined as,

$$b(\omega, \phi) = \int_\Omega \nabla\omega \cdot \left(f_r^{-1} \nabla \phi\right) d\Omega - k_0^2 g_r \int_\Omega \omega \phi \, d\Omega + j k_0 f_r^{-1} \int_{\Gamma_S} \omega \phi \, d\Gamma \tag{6}$$

and linear form f is defined as follows,

$$f(\omega) = -\underbrace{\int_\Omega \omega q \, d\Omega}_{b_q} + \underbrace{\int_{\Gamma_S} \omega \Psi \, d\Gamma}_{b_\Psi}. \tag{7}$$

Note that PEC and PMC are implemented as homogeneous Neumann boundaries for the TE and TM cases, respectively. Thus, the boundary terms are omitted in (6). The excitation of the problem may come from internal sources (term q of the formulation corresponding to impressed currents J_z or M_z) or by exterior source determined by residual function Ψ of B.C. associated to exterior boundary S.

Variational formulation (5)–(7) is discretized by proper choice of discrete hp-versions of space H_0^1. Hierarchical shape functions defined over quadrilateral finite elements are used to span subspaces of H^1. Basis functions are naturally associated to vertexes, edges and interior of the element. The restriction to H_0^1 is made simply by setting to zero the degrees of freedom associated to vertexes and edges of the Dirichlet boundaries.

Shape functions are defined on the master/reference quadrangle (of side length equal to one, Fig. 3) as the tensor product of one dimensional (1D) shape functions $\hat{\chi}_i(t),\ t \in [0, 1]$ such that

$$\hat{\chi}_1(t) = 1 - t \tag{8}$$

$$\hat{\chi}_2(t) = t \tag{9}$$

and

$$\hat{\chi}_j(t) \in \tilde{P}_j([0, 1]) \quad \text{satisfying } \hat{\chi}_j(0) = \hat{\chi}_j(1) = 0 \quad \text{for } j = 3, \ldots, p+1 \tag{10}$$

where $\tilde{P}_j$ denotes the space of homogeneous polynomials of order j. Obviously, it is required $span\{\hat{\chi}_1(t), \hat{\chi}_2(t), \hat{\chi}_3(t), \ldots, \hat{\chi}_{p+1}(t)\} = P_p([0, 1])$ with $P_p([0, 1])$ used to refer to the space of polynomials of order p defined on the interval $[0, 1]$.

There is one shape function associated to each one of the four vertexes for the master quadrangle:

$$\hat{\phi}_1(\xi,\eta) = \hat{\chi}_1(\xi)\hat{\chi}_1(\eta) = (1-\xi)(1-\eta) \tag{11}$$

$$\hat{\phi}_2(\xi,\eta) = \hat{\chi}_2(\xi)\hat{\chi}_1(\eta) = \xi(1-\eta) \tag{12}$$

$$\hat{\phi}_3(\xi,\eta) = \hat{\chi}_2(\xi)\hat{\chi}_2(\eta) = \xi\eta \tag{13}$$

$$\hat{\phi}_4(\xi,\eta) = \hat{\chi}_1(\xi)\hat{\chi}_2(\eta) = (1-\xi)\eta \tag{14}$$

$p_{ek} - 1$ shape functions per each one of the four edges ($k = 1, \ldots, 4$):

$$\hat{\phi}_{5,j}(\xi,\eta) = \quad \hat{\chi}_{2+j}(\xi)\hat{\chi}_1(\eta) \quad j = 1, \ldots, p_{e1} - 1 \tag{15}$$

$$\hat{\phi}_{6,j}(\xi,\eta) = \quad \hat{\chi}_2(\xi)\hat{\chi}_{2+j}(\eta) \quad j = 1, \ldots, p_{e2} - 1 \tag{16}$$

$$\hat{\phi}_{7,j}(\xi,\eta) = \hat{\chi}_{2+j}(1-\xi)\hat{\chi}_2(\eta) \quad j = 1, \ldots, p_{e3} - 1 \tag{17}$$

$$\hat{\phi}_{8,j}(\xi,\eta) = \hat{\chi}_1(\xi)\hat{\chi}_{2+j}(1-\eta) \quad j = 1, \ldots, p_{e4} - 1 \tag{18}$$

and $(p_h - 1)(p_v - 1)$ interior shape functions

$$\hat{\phi}_{9,i,j}(\xi,\eta) = \hat{\chi}_{2+i}(\xi)\hat{\chi}_{2+j}(\eta) \quad i = 1, \ldots, p_h - 1,\ j = 1, \ldots, p_v - 1. \tag{19}$$

Note that different orders may be chosen for edges and interiors by proper setting of p_{e1}, p_{e2}, p_{e3}, p_{e4}, p_h, p_v. Furthermore, anisotropic orders are supported, i.e., different orders in local "horizontal" and "vertical" directions of the element, i.e., by selecting $p_h \neq p_v$. The orders of the edges and interior are requested to satisfy the *minimum rule*, i.e., $p_{e1}, p_{e3} \leq p_h$ and $p_{e2}, p_{e4} \leq p_v$. Thus, the functions described above span space $X(\hat{K})$ given by

$$X(\hat{K}) = \{\hat{u} \in Q^{(p_h,p_v)};\ u|_{\hat{e}_i} \in P^{p_{ei}}(\hat{e}_i)\} \tag{20}$$

where $\hat{K}$ denotes the master quadrangle, $\hat{e}_i$ the ith edge of the master quadrangle and $Q^{(p_h,p_v)} = P^{p_h} \otimes P^{p_v}$, i.e., polynomials that are of order p_h in ξ and order p_v in η.

Several 1D shape functions $\hat{\chi}_i(t)$ are coded. Specifically, the results shown in this paper have been obtained using integrated Legendre polynomials as higher order shape functions, i.e.,

$$\hat{\chi}_j(t) = \int_0^t L_{j-2}(t)\,dt, \quad j = 3, \ldots \tag{21}$$

where $L_{j-2}(t)$ are shifted Legendre polynomials (orthogonal in interval [0, 1]).

Shape functions on the *real* element are obtained by transformation from the master element. Curved elements are supported through the concept of isoparametric elements. In terms of discretization, the code supports irregular meshes (i.e., with *hanging nodes*) by imposing linear combination of the edge shape functions involved in the assembly of the finite elements. Specifically, the meshes supported are 1-irregular, i.e., the parent element of a given element of a refined mesh does not have hanging nodes. Further details can be found in [1].

The field in Ω^{FEM} can be obtained as the FEM solution of (5), provided that the residual of the boundary condition on S (Γ_S), i.e., Ψ, is known. The value of Ψ cannot be known before solving the problem; however, it can be estimated from the exterior problem. By means of the Equivalence Principle (see e.g., [34, Section 3–5]), the exterior domain Ω^{EXT} is electromagnetically modeled with the IE of the equivalent problem exterior to S' using the equivalent currents $\mathbf{J}_{\text{eq}}$ and $\mathbf{M}_{\text{eq}}$ of Ω^{FEM} (computed by means of the FEM solution), and the Green's function G of Ω^{EXT} (a zero order second class Hankel function for the free-space case). The scattered (or radiated) field produced by equivalent currents will be denoted by symbol ϕ_{sc}. The total field is obtained by adding the scattered field ϕ_{sc} to the excited incident field ϕ_{inc} (see Fig. 1). The scattered (or radiated field) ϕ_{sc} is obtained from equivalent currents as:

$$\phi_{\text{sc}}(\boldsymbol{\rho}) = \oint_{S'} \left[L^t_{\text{eq}}(\boldsymbol{\rho}') \frac{\partial\, G(\boldsymbol{\rho}, \boldsymbol{\rho}')}{\partial n'} - jk_0 I_0 O^z_{\text{eq}}(\boldsymbol{\rho}')\, G(\boldsymbol{\rho}, \boldsymbol{\rho}') \right] dl' \quad \boldsymbol{\rho} \in \Gamma_S \tag{22}$$

and its derivative,

$$\frac{\phi_{\text{sc}}(\boldsymbol{\rho})}{d\rho} = \oint_{S'} \left[-jk_0 I_0 O^z_{\text{eq}}(\boldsymbol{\rho}') \frac{\partial\, G(\boldsymbol{\rho}, \boldsymbol{\rho}')}{\partial n} + L^t_{\text{eq}}(\boldsymbol{\rho}') \frac{\partial}{\partial n} \left(\frac{\partial\, G(\boldsymbol{\rho}, \boldsymbol{\rho}')}{\partial n'} \right) \right] dl' \quad \boldsymbol{\rho} \in \Gamma_S \tag{23}$$

where the superindexes t and z refer to the tangential and longitudinal components, respectively. The equivalent currents are defined in terms of ϕ, or its normal derivative, according to Table 1.

The total field ϕ calculated following this procedure, and its normal derivative, are plugged into (4) to obtain the new value for Ψ. Note that the solution of the interior problem is needed in order to find $\mathbf{J}_{\text{eq}}$ and $\mathbf{M}_{\text{eq}}$. Thus, the entire problem is solved in an iterative fashion resembling domain decomposition Schwarz iterations (see Fig. 2). An initial value of Ψ must be selected at the first iteration. A natural choice is $\Psi^{(0)}_{\text{sc}} = 0$, i.e., $\Psi^{(0)} = \Psi_{\text{inc}}$ for scattering problems and $\Psi^{(0)} = 0$ for radiation

problems. Function Ψ_{inc} is obtained by evaluating left side of (4) with $\phi = \phi_{\text{inc}}$. FE-IIEE methodology (in its ACA accelerated version) is described as Pseudocode 1. Pseudocode corresponding to the entire flow of the self-adaptive *hp* methodology presented in Fig. 2 is described as Pseudocode 2.

Algorithm 1 FE-IIEE with ACA acceleration

```
procedure FE-IIEE_ACA(mesh,K,b,U,φ)

    Clusterize mesh (and hence S, S') into blocks: Z = Σ_b Z_b
    for 1 ... number_of_Zb do
        blocks_are_far ← determine if Z_b is a far-off interaction
        if blocks_are_far then
            Use ACA to compute U, V: Z_b ≈ U V                ▷ Expression (28)
        else
            Compute Z_b directly using (25)–(27)
        end if
    end for

    error_gt_U ← true
    i ← 0
    b^(0) = b
    φ^(0) ← solve(K, b^(0))                    ▷ K is FEM matrix from discretization of (6)
    while error_gt_U do
        b^(i+1) ← Update RHS using Z, φ^(i) and b^(0)          ▷ Expression (24)
        φ^(i+1) ← solve(K, b^(i+1))
        error ← norm(φ^(i+1) − φ^(i))
        error_gt_U ← compare error with U
        i ← i + 1
    end while

                                   ▷ Return of last solution φ and RHS b
    φ ← φ^(i)
    b ← b^(i)

end procedure
```

When Ψ is updated at each FE-IIEE iteration, the substitution of (22), (23) into (4) (updated Ψ), and, in turn, the introduction of (4) (Ψ) into the second term of (7) leads to a double spatial integration on S' and S, namely, b_Ψ. This convolution takes into account the interactions between the equivalent sources on every point on S' and the test function on every point on S (or the field radiated by $\mathbf{J}_{\text{eq}}$ and $\mathbf{M}_{\text{eq}}$ on S', evaluated on the S boundary and weighted by the test function ω), and has a computational complexity $O(N^2)$ for each iteration of the FE-IIEE algorithm, being N the number of unknowns placed at S. For simplicity in our explanations we assume $N \approx N'$ where N' refers to the unknowns associated to S'. The described procedure to update b_Ψ in the ith iteration may be compactly expressed in matrix form as follows:

$$\{\mathbf{b}_\Psi^{(i+1)}\} = \mathbf{C}\,\{\Psi^{(i+1)}\} = \mathbf{C}\,\{\Psi^{(0)}\} + \mathbf{C}\mathbf{M}\left\{\mathbf{u}_{S'}^{(i)}\right\} = \mathbf{C}\,\{\Psi^{(0)}\} + \mathbf{Z}\left\{\mathbf{u}_{S'}^{(i)}\right\} \tag{24}$$

where $\mathbf{C}$ is a rectangular matrix that weights the residual vector Ψ with the interpolating functions on S (term b_Ψ of the RHS of (7)) and $\mathbf{M}$ represents the integral operator given by the Green's function (expressions (22), (23)) together with the action of combining these expressions into (4). Finally, $\mathbf{u}_{S'}^{(i)}$ is the vector of the degrees of freedom (dof) previously found in the preceding FE-IIEE iteration. That is, matrix $\mathbf{Z}$ is of the form

$$\mathbf{Z} = \mathbf{Z}_{d\phi} + jk_0\mathbf{Z}_\phi \tag{25}$$

where the element of the i row and the j column of the matrices above:

$$z_{\phi_{ij}} = \int_{\Gamma_S} \omega_i(\boldsymbol{\rho}) \oint_{S'} \left(\omega'_j(\boldsymbol{\rho}') \frac{\partial G(\boldsymbol{\rho}, \boldsymbol{\rho}')}{\partial n'} - \frac{\partial \omega'_j(\boldsymbol{\rho}')}{\partial n'} G(\boldsymbol{\rho}, \boldsymbol{\rho}') \right) dl'\, d\Gamma \tag{26}$$

$$z_{d\phi_{ij}} = \int_{\Gamma_S} \omega_i(\boldsymbol{\rho}) \oint_{S'} \left(-\frac{\partial \omega'_j(\boldsymbol{\rho}')}{\partial n'} \frac{\partial G(\boldsymbol{\rho}, \boldsymbol{\rho}')}{\partial n'} + \omega'_j(\boldsymbol{\rho}') \frac{\partial^2 G(\boldsymbol{\rho}, \boldsymbol{\rho}')}{\partial n\, \partial n'} \right) dl'\, d\Gamma \tag{27}$$

where ω_i represents any of the basis functions defined on the boundary S, i.e., those basis functions corresponding to the nodes of the element edges lying on S; and ω'_j corresponds to the basis functions associated with the elements touching

Algorithm 2 Self-adaptive *hp*-FEM

1: Read input data (geometry, b.c., material constants, excitations, T)

2: coarse_mesh ← initial mesh generation from geometry
3: $\mathbf{K}_C$, $\mathbf{b}_C^{(0)}$ ← Assembly of finite element matrices and rhs

4: U_C ← Set U for initial mesh
5: $\mathbf{b}_C$ ← $\mathbf{b}_C^{(0)}$
6: error_hp_gt_T ← true
7: **while** error_hp_gt_T **do**
8: ϕ_C ← FE-IIEE_ACA(coarse_mesh,$\mathbf{K}$,$\mathbf{b}_C$,U_C,ϕ_C)

9: fine_mesh ← Uniform refinement in h and p of coarse_mesh
10: U_F ← Set U for fine mesh ▷ Typically $U_F = U_C/10$
11: $\mathbf{K}_F$, $\mathbf{b}_F^{(0)}$ ← Assembly of finite element matrices and rhs
12: $\mathbf{b}_F$ ← RHS update using $\mathbf{b}_F^{(0)}$ and projection of $\mathbf{b}_C$ on mesh_fine
▷ Note that $\mathbf{b}_C$ was updated in line 8
13: ϕ_F ← FE-IIEE_ACA(fine_mesh,$\mathbf{K}_F$,$\mathbf{b}_F$,U_F,ϕ_F)

14: error_function ← $(\phi_F - \phi_C)$
15: error_hp ← norm(error_function) ▷ error_hp refers to e_{hp}
16: error_hp_gt_T ← compare *error_hp* with T
17: **if** error_hp_gt_T **then**
18: U_C ← Set U for next coarse mesh ▷ Typically $U_C = e_{hp}/10$
19: mesh_coarse ← NewOptimalMesh(coarse_mesh,geometry,error_function)
20: $\mathbf{K}_C$, $\mathbf{b}_C^{(0)}$ ← Assembly of finite element matrices and rhs
21: $\mathbf{b}_C$ ← RHS update using $\mathbf{b}_C^{(0)}$ and projection of $\mathbf{b}_F$ on coarse_mesh
22: **end if**
23: **end while**

▷ Note FE-IIEE_ACA procedure is defined in Pseudocode 1
▷ Details about NewOptimalMesh procedure are discussed in the paper (see also [1, 30]).

the inner bound, S'. Note that, unlike the case of ω_i, the derivative of ω'_j is part of the formulation. Since the derivatives of the basis functions associated to the inner nodes of the FEM elements are non-zero on the edges, it occurs that not only the basis functions corresponding to the element edges placed on S' has to be considered, but every basis function defined for the entire element (with one or more edges lying on S') has to be taken into account. Thus, matrix $\mathbf{Z}$ is similar to a Method of Moment (MoM, [35]) matrix but with different number of rows and columns due to the considerations just mentioned above.

In the conventional version of the method (i.e., without any type of acceleration with ACA), matrix $\mathbf{Z}$ is not calculated explicitly but the scattered field (and its derivative) in the integration points on S is calculated using (22), (23) by accumulating all contributions from field (and its derivative) at integration points over finite elements on S'. Thus, the weighting procedure to obtain a given coefficient of $\{\mathbf{b}_\Psi\}$ does not involve explicit computation of all coefficients of a given column of $\mathbf{Z}$. In other words, the results of the matrix vector multiplication involving $\mathbf{Z}$ and $\mathbf{u}_{S'}$ (see the last term of (24)) are basically implicit when using directly (22), (23).

When the method is accelerated by using ACA, coefficients of matrix $\mathbf{Z}$ are explicitly calculated using (26), (27). However, when ACA comes into play only a small number of rows and columns of $\mathbf{Z}$ need to be computed with considerable saving in CPU time and storage requirements, as it is explained below. This is possible due to the numerically rank-deficient nature of the far-off interactions, in which mutual electromagnetic influence is weak enough to substantially compress the information relative to such interaction while achieving a given precision.

In order to apply ACA, the entire FEM domain is clusterized in a FMM fashion. This is equivalent to split the matrix $\mathbf{Z}$ into several sub-matrices, namely $\mathbf{Z_b}$, each one corresponding to the electromagnetic interaction between two given physical regions (groups) placed on the boundaries S and S'. The sub-matrices corresponding to the interaction between two clusters separated by the distance of one or more ACA blocks, can be compressed up to a given accuracy level by its factorized form as follows:

$$\mathbf{Z_b}^{m\times n} \approx \widetilde{\mathbf{Z}}_\mathbf{b}^{m\times n} = \mathbf{U}^{m\times r}\,\mathbf{V}^{r\times n} = \sum_{i=1}^{r} \mathbf{U}_i^{m\times 1}\,\mathbf{V}_i^{1\times n}. \tag{28}$$

The superscripts refer to the dimensions of the matrices and vectors in the equation above. $\mathbf{U}_i^{m\times 1}$ and $\mathbf{V}_i^{1\times n}$ represent the ith row and the ith column of the decomposition matrices $\mathbf{U}$–$\mathbf{V}$. Superscript m refers to the number of dof (block wise) associated to boundary S while superscript n refers to the dof of finite elements (block wise) touching boundary S'. Note that one of the dimensions of the approximated sub-matrix has been reduced to r, which is the numerical rank of the compressed $\mathbf{Z_b}$. Since ACA is a rank-revealing decomposition, it provides the algorithm to efficiently compute the $\mathbf{U}$ and $\mathbf{V}$ matrices on the fly, i.e., without knowing the coefficients of $\mathbf{Z_b}$ or the numerical rank in advance. Specifically, the resultant r is the number of rows and columns of the original sub-matrix that have to be computed in order to yield a given accuracy. Specifically, the error criterion for each sub-block is controlled by only one parameter, which is common to all sub-matrices, denoted as ϵ:

$$E_b = \|\mathbf{Z_b}^{m\times n} - \widetilde{\mathbf{Z_b}}^{m\times n}\| \leq \epsilon \|\mathbf{Z_b}^{m\times n}\| \tag{29}$$

being E_b the error in the approximation of the corresponding sub-matrix and $\|.\|$ denoting the Frobenius norm.

It is worth noting that when the $\mathbf{Z}$ matrix comes into play and is approximated ACA version, the sub-matrices of $\mathbf{Z}$ relating two distant groups need to be calculated only once when processing either the coarse mesh or the fine mesh of the present hp iteration, being $\mathbf{U}$–$\mathbf{V}$ stored through the FE-IIEE process, and multiplied by the vector of updated dof on S', $\mathbf{u}_{S'}^{(i)}$ at each FE-IIEE iteration.

4. Computational complexity estimates

In this section, estimates corresponding to the computational complexity (in terms of number of computer operations) of the method presented in the paper are shown; specifically those corresponding to the computation of the FE-IIEE iterations (inner loop of Fig. 2). That is, estimates presented below correspond to the computations performed to obtain the value of $\{\mathbf{b}_\Psi^{(i+1)}\}$ from FEM field solution associated to S' at ith iteration ($\mathbf{u}_{S'}^{(i)}$, box highlighted in red in Fig. 2). Actually, the estimates correspond to the accumulated computations along all iterations performed by FE-IIEE loop. In the following we will use I to denote the number of iterations of FE-IIEE.

As it was mentioned in the previous section, there are two ways to perform those computations: conventional way without ACA and accelerated way using ACA. In the conventional way, scattered field on S (and its derivative) is calculated using (22), (23) by accumulating all contributions from field associated to S' without explicit computation of any matrix coefficients. Nevertheless, the double loop type of operation behind (22), (23) to compute the scattered field (and its derivative) together with their integral over S to get $\{\mathbf{b}_\Psi\}$ provide a dominant complexity for the computer time in terms of the square of number of unknowns N associated to the boundary. If we denote by t_{INT} the time involved in such computation, it can be demonstrated that

$$t_{\mathrm{INT}} \propto (N^2(2p+16) + 3N(p+1)) \cdot I \tag{30}$$

where we observe the quadratic dependence of t_{INT} with the number of unknowns on the boundaries. Note that the term p is routinely discounted in specialized literature because polynomial remains constant in customary FEM and MoM approaches.

When FE-IEEE computations are accelerated by using ACA, matrix $\mathbf{Z}$ coefficients are implicitly calculated according to (25)–(27) and $\{\mathbf{b}_\Psi\}$ at each iteration is obtained by means of matrix–vector multiplication (see (24)). The acceleration comes from the fact that most of the sub-matrices $\mathbf{Z_b}$ of $\mathbf{Z}$ that appear after partitioning of boundaries S and S' in blocks may be compressed on the fly and stored in much smaller row-type $\mathbf{U}$ and column-type $\mathbf{V}$ matrices (see (28)). Once $\mathbf{U}$ and $\mathbf{V}$ matrices for interactions between pairs of "far" block interactions are computed and stored, matrix–vector operations with finite element degrees of freedom based on factorized representation of sub-matrices $\mathbf{Z_b}$ as in (28) are performed. We will use t_{UV} to refer to the time corresponding to the computation of coefficients of matrices $\mathbf{U}$, $\mathbf{V}$ and t_{mult} to the one corresponding to the matrix–vector multiplication for one iteration of FE-IIEE. Thus, the time consumed in FE-IIEE iterations when ACA is used (denoted by t_{ACA}) is: $t_{\mathrm{ACA}} = t_{\mathrm{UV}} + t_{\mathrm{mult}} \cdot I$.

Time t_{ACA} in turn includes, on one hand, the calculus of the *selected* exact terms of the original impedance matrix $\mathbf{Z}$ to be approximated, whose complexity can be shown to be proportional to $G_{\mathrm{f}}\, rN(12p^3 + 39p^2 + 57p + 30)$, where G_{f} is the medium number of far-off ACA groups and r is the medium numerical rank of the associated sub-matrices $\mathbf{Z_b}$. Note the cubic dependence on p in this case compared with linear dependency in the case of the straightforward integration corresponding to (30). And, on the other hand, the time expended to perform the operations corresponding to the iterative ACA algorithm itself must be also included. This time scales as $G_{\mathrm{f}}\, r^2N(2p+1)$. Thus, scalability of t_{UV} goes as

$$t_{\mathrm{UV}} \propto O(\underbrace{G_{\mathrm{f}}\, rN(12p^3 + 39p^2 + 57p + 30)}_{\text{Computation of } G_{\mathrm{f}}rN \text{ original elements of Z}} + \underbrace{G_{\mathrm{f}}\, r^2N(2p+4)}_{\text{ACA loop cost}}). \tag{31}$$

The time expended in the matrix–vector multiplication of the ACA compressed matrix with vector degrees of freedom scales as

$$t_{\mathrm{mult}} \propto O(\underbrace{2G_{\mathrm{f}}\, rN(p+1)}_{\mathbf{U}\cdot\mathbf{V}\cdot\mathbf{u}'}) \tag{32}$$

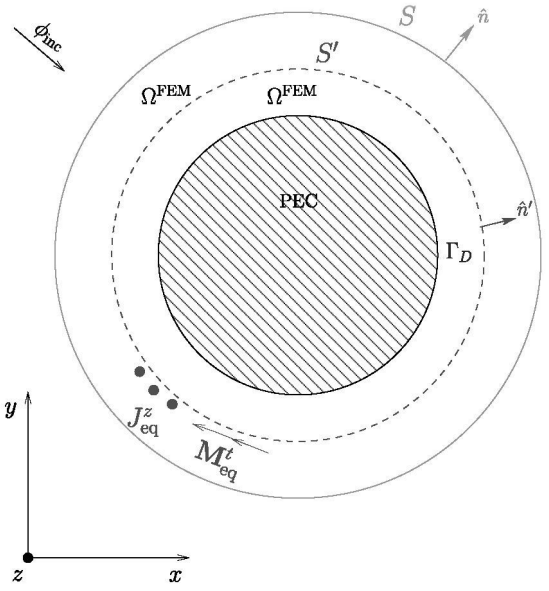

Fig. 4. FE-IIEE setup for the scattering problem of a circular PEC with TM polarization (E_z formulation).

and, hence,

$$t_{\text{ACA}} \propto \underbrace{G_{\text{f}}\, rN(12p^3 + 51p^2 + 69p + 30) + G_{\text{f}}\, r^2N(2p+4)}_{t_{\text{UV}}} + \underbrace{2G_{\text{f}}\, rN(p+1)I}_{t_{\text{mult}\cdot I}}. \tag{33}$$

It is important to note that the speed-up provided by ACA acceleration comes from the fact that the original square dependency with the number of unknowns, i.e., N^2, of the convolutional character of the computation of the scattered field gets reduced to $G_{\text{f}}\, rN$. Thus, if $\mathbf{Z}$ matrix would have been calculated without any use of ACA, which is the case for MoM method (or its hybridized version FEM-BEM), we would have obtained

$$t_{\text{MoM}} \propto \underbrace{N^2(12p^3 + 51p^2 + 69p + 30) + N^2(2p+4)}_{t_{\text{Z}}} + \underbrace{N^2(p+1)I}_{t_{\text{multZ}\cdot I}}. \tag{34}$$

Expression (34) is just shown here to explicitly illustrate where ACA acceleration comes into play; however, in the non-accelerated version of the approach presented here there is not any explicit computation (or storage) of any $\mathbf{Z}$ and its associated computational time corresponds to expression (30). Thus, the appropriate comparison in terms of performance of the methods is between those corresponding to expressions (33) and (30). The numerical results shown in Section 5.2 correspond precisely to this comparison between (33) and (30).

5. Numerical results

5.1. Verification & validation

Numerical results obtained from the application of the ACA algorithm to the integral representation of the exterior field in the context of *hp*-automatic adaptivity for open domains are presented next. Although the implementation of the methodology shown in the previous section is valid for both scattering and radiation problems with arbitrary composition of materials and geometries, the results of this paper correspond to the scattering of plane waves on perfect conductor bodies. Specifically, several infinitely long cylindrical scatterers with perfect electric conducting (PEC) properties have been successfully analyzed. The incidence direction of the plane wave illuminating the object is 0° in every case. Very coarse meshes of second order ($p = 2$) are used as initial grids in order to assess the robustness of the *hp* strategy when combined with ACA as acceleration method for the truncation of the domain.

In order to validate the implementation, a problem with known analytical solution is considered first. It consists of the scattering of an incident plane wave on a PEC circular shape cylinder of radius R equal to 5λ where symbol λ denotes the wavelength, i.e., $\lambda = 2\pi/k_0$. Polarization of the wave is TM (formulation is with E_z). The setup of the problem is shown in Fig. 4. As the scatterer is PEC, the corresponding B.C. for the TM case is of homogeneous Dirichlet type (2). According to FE-IIEE methodology, a contour line S' has to be chosen that encloses the scatterer. The most efficient option in terms of the number of unknowns is to choose S' conformal to the scatterer boundary. Thus, contour line S' is chosen to be circular. Boundary S' can be placed on top of the scatterer surface if desired. This is a good choice in practice because it minimizes

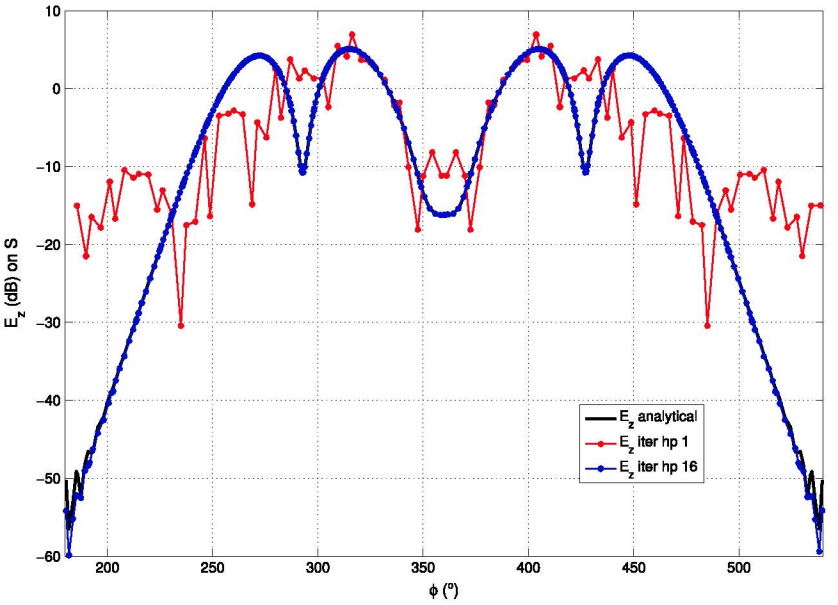

Fig. 5. Comparison of field on S obtained with hp-ACA and analytic solution. ACA parameter $\epsilon = 0.01$. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

the number of unknowns. Furthermore, in the particular case of a PEC object some of the equivalent current source terms in (22), (23) (and hence, (26), (27)) would be null. Specifically, for TM polarization PEC is a Dirichlet contour making $\mathbf{M}_{\text{eq}} = 0$ on the PEC contour. Although desirable in practice for PEC scatterers, we have preferred not to place S' on top of the PEC scatterer but at some distance so all equivalent current terms involved in the integral equation are active to resemble the setup in the general case. The mesh truncation contour S has to be chosen so that it encloses S at some distance. Here, it is chosen to be conformal to the scatterer. The distance from the scatterer to the conformal boundary S' is 0.5λ, and the same distance is left between S' and S. It is worth noting that the distance from S' to S in FE-IIEE is usually small, typically in the range of 0.05λ to 0.2λ. However, a larger gap has been chosen for representation purposes. As the object is not permeable to the electromagnetic field, the representation of the final mesh delivered by the automatic adaptivity in a practical case would have been too small to be illustrative.

A comparison of the magnitude of the obtained field (E_z component) on the exterior boundary S with the analytical solution is shown in Fig. 5. The field levels for the first hp iteration (red color) and the 16th hp iteration (blue color) are shown. The convergence of the method from a rough initial solution to the analytical solution (black color) can be observed. The hp-mesh delivered after 17 iterations is shown in Fig. 6. Different colors indicate the order of approximation p of the elements.

The exact error of the FEM field solution on S after 16 hp iterations is about 0.04%, which is the same error obtained without ACA acceleration. That is, there is no impact in the sequence of meshes delivered by the hp-adaptivity because of ACA; specifically, the final mesh of Fig. 6 is identical to the mesh obtained without ACA. The threshold error level for the ACA algorithm was set to $\epsilon = 0.01$. This value in this problem does not influence the hp strategy and the error level achieved, as it will be clear later. In contrast, a great amount of computational saving is achieved even with this very demanding value of ϵ. In this example, a quarter of the operations needed to compute the boundary condition on S were carried out when ACA was used in comparison with the computation of the IE without ACA.

As an example of scattering problem with field singularities, a PEC square scatterer of side $L = 2\lambda$ has been considered next. An analogous setup to the one of the circular scatterer is made here. Conformal contours S and S' are chosen, i.e., of rectangular shape. In this case, the boundary S is placed to a distance 0.1λ and 0.2λ from S' and from the scatterer respectively. The field obtained after one hp iteration and the field solution on S after 17 hp iterations are shown in Fig. 7. The field solution after 40 hp iterations has been taken as the reference field solution to compute the error. It can be seen that no detectable improvement is obtained for the FEM solution after 17 iterations. The energy norm of the error between the coarse grid solution and the fine grid solution (e_{hp}) for 1, 17 and 40 hp iterations is 16%, 0.63% and 0.015%, respectively. The mesh corresponding to the 17th iteration of the self-adaptive strategy is shown in Fig. 8. As expected, h refinements close to the corners or field singularities and p refinements where the field variation is smooth, are observed. As it was mentioned above, the final mesh along with the error obtained are not influenced by the ACA algorithm, provided that ACA error parameter ϵ is chosen of the same order of magnitude than the desired estimated energy error. Specifically, in this case, a value of $\epsilon = 0.1\%$ was used, which is one order of magnitude lower than the threshold hp error $T = 1\%$, and of the same order of magnitude than the actual estimated error achieved, $e_{hp} = 0.63\%$.

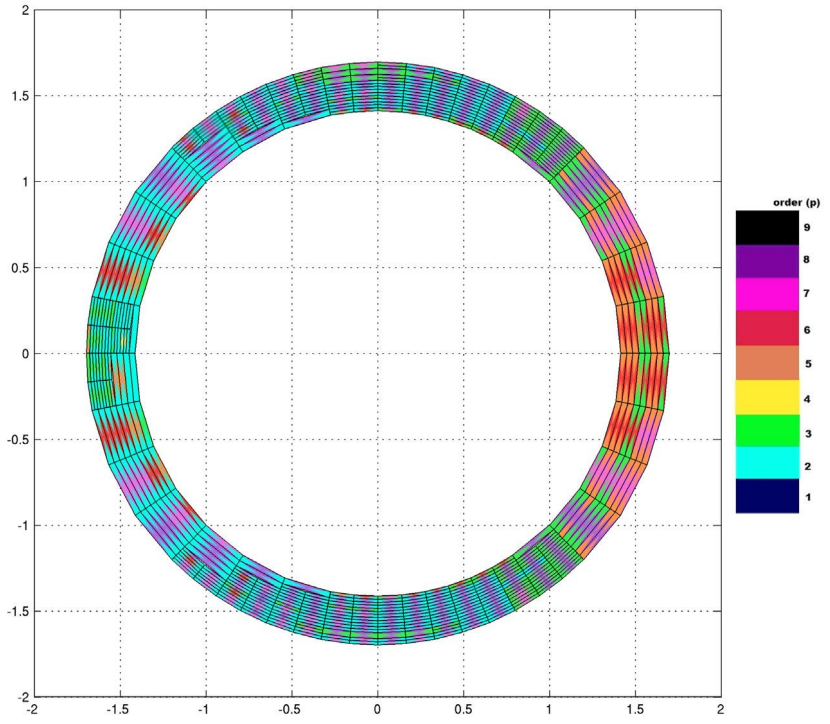

Fig. 6. Self-adaptive *hp*-mesh after 16 iterations. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

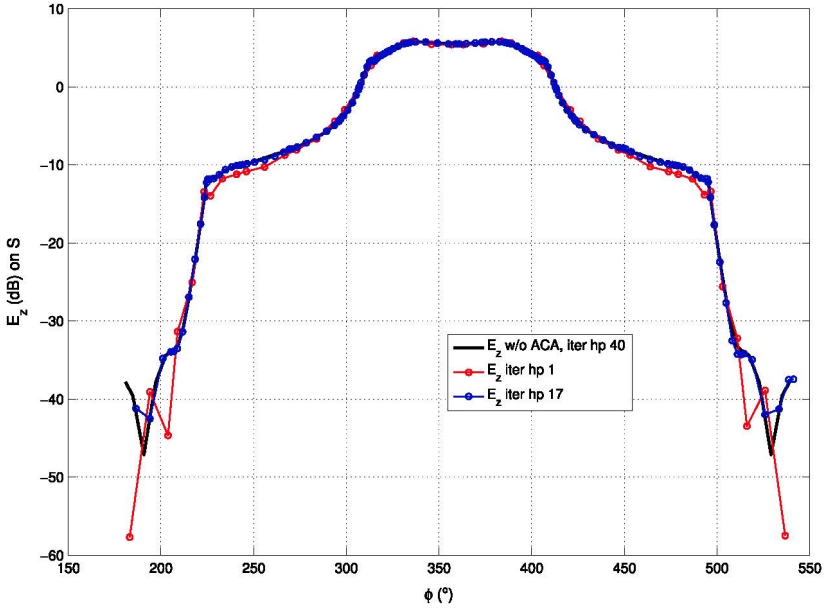

Fig. 7. Comparison of field on S obtained with *hp*-ACA and a reference solution. ACA parameter $\epsilon = 0.1$.

In order to prove the robustness of the ACA-*hp* method, the impact on the accuracy of the ϵ parameter in this context, as well as to measure its influence on FEM-*hp* performance, the exact error of the field on S versus the value of ϵ has been measured for *hp*-meshes corresponding to different iterations of the adaptivity. The case of the circular PEC scatterer has been considered. The results corresponding to meshes delivered after a given relevant number of *hp* iterations are presented in Fig. 9. The compression obtained by ACA for the $\mathbf{Z}$ matrix for each one of the points depicted in Fig. 9 is shown in Fig. 10.

Computational resources (memory and, hence, CPU time) are reduced as long as ϵ is increased (see Fig. 10). This computational saving reaches a physical limit for a certain maximum value of ϵ (10^0 in this case) and the error remains

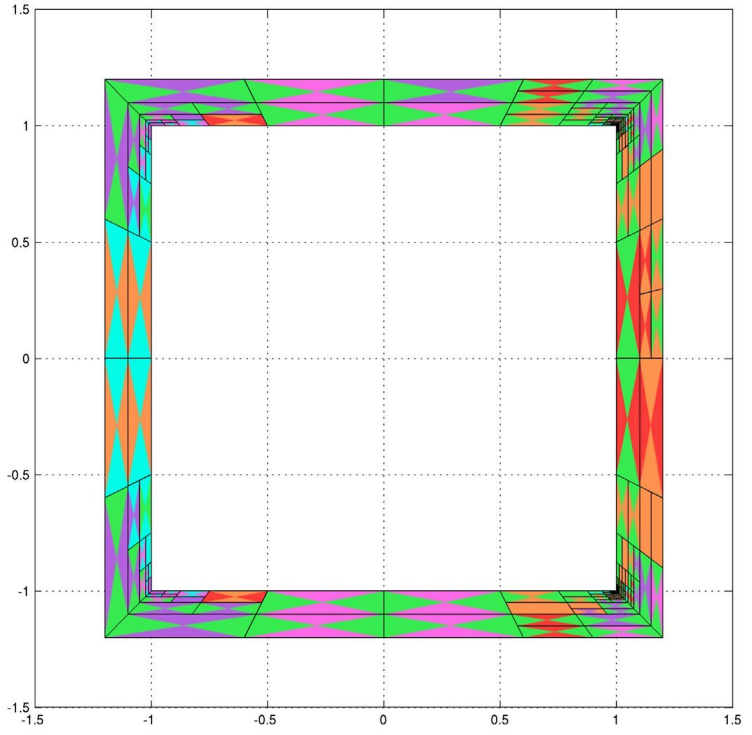

Fig. 8. Self-adaptive *hp* mesh after 17 iterations.

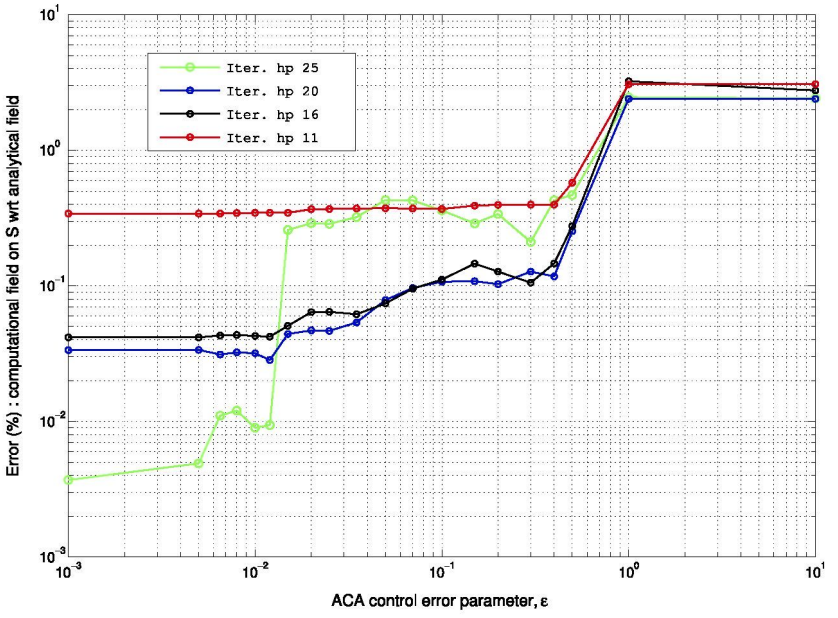

Fig. 9. Exact error of the field on S versus ϵ. Different curves correspond to different *hp*-iterations as indicated in the legend, i.e., to different values of threshold T. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

constant (right part of the graphic of Fig. 9). That is, for large values of ϵ, the accuracy is limited by ACA and not the *hp* discretization. This is due to the fact that only one row and one column of $\mathbf{Z}$ are required by ACA to be computed and this is the absolute minimum number of coefficients achievable. At this point, maximum compression and error are obtained. It is worth noting that the error does not exceed a certain level (around 3% in this example) even with $\epsilon \to \infty$, which demonstrates the robustness of the ACA algorithm. However, for certain extreme cases (very large values of ϵ compared with T), the error introduced by ACA in the FE-IIEE loop may lead to a misguidance of the refinement strategy. As the value

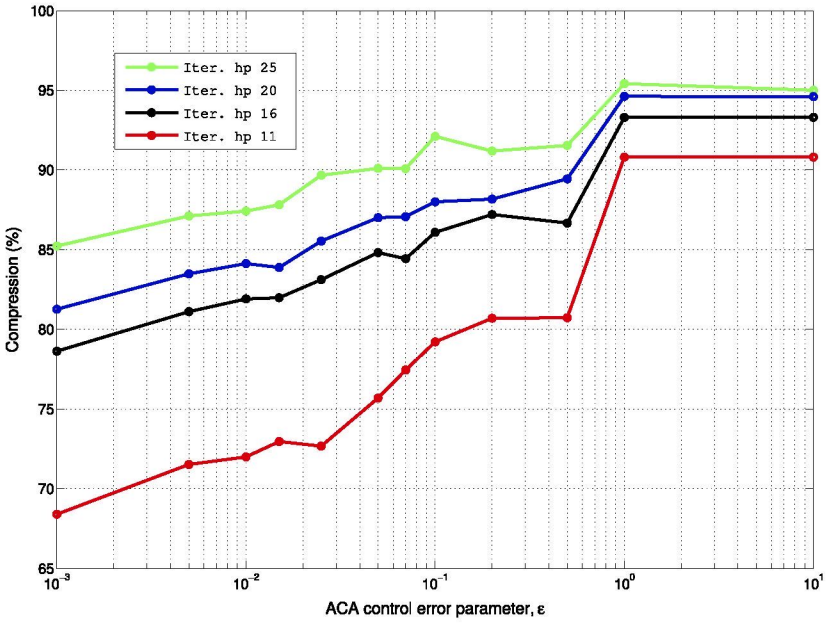

Fig. 10. ACA compression versus ϵ. Same conditions of the experiment as in Fig. 9 hold.

of ϵ is decreasing the limitation in the accuracy provided by the ACA approximation has a lower impact in the field error. This is clearly noticed in the left part of the graphic of Fig. 9 in which the very low value of ϵ (around 10^{-3}) assures the error is limited by the *hp* discretization and not by ACA. It is observed that *hp*-meshes corresponding to a higher number of iterations (and hence, lower error threshold T demanded) provide lower field errors as ACA is not limiting the accuracy. In practical terms, this error floor is translated to values of ϵ in the range of $[T - T/10]$. Therefore, below this value of ϵ no gain is obtained by further decreasing the ACA threshold ϵ; instead, more computational resources are consumed and wasted (time and memory). Nevertheless, above this value of ϵ, computational savings can be gained at the expense of limiting (in a controlled manner) the accuracy of the *hp* methodology that, at the end, is the main goal behind the use of ACA. There is an intermediate zone characterized by a sort of proportional relation between ϵ and the error (and specifically, of the same order of magnitude), i.e., the error is still limited by the ACA approximation. The behavior of the case of "25 *hp*-iterations" (curve on green color in the figure) differs from the behavior of the other cases in this intermediate zone being an example of misguidance of the refinement strategy as it was mentioned above. The reason is that the ACA parameter ϵ is so large compared with the error level T requested that the estimation of the error function used to lead the self-adaptive *hp*-algorithm fails.

With respect to compression levels, it is observed in Fig. 10 that compressions above 70% are easily obtained within the optimum range for ϵ. Furthermore, the compression variation is similar for every iteration descending uniformly independently of the level of accuracy required (value of ϵ) and the level of refinement.

For a fixed value of ϵ, the compression percentage increases noticeably as the number of unknowns in the mesh grows. The more demanding the mesh, the more compression obtained. With the meshes obtained after 25 *hp*-iterations, compressions of 85% as minimum are achieved. This is due to the redundancy in the information: as long as the automatic refinement process takes place and N and the accuracy are increased, the rank rises far more slowly than the number of rows in the matrix $\mathbf{Z}$ does (see examples of Section 5.2). Nevertheless, once the minimum effective ϵ is reached, namely, that ϵ below for which the accuracy cannot be improved, the accuracy stagnates (as it was commented above) but the compression continues being reduced.

Thus, it can be concluded that there exists a range of "optimum" values for ϵ being in the order of $T/10$ as the lowest value, which fulfills the minimum error and maximum compression level without having an impact on the accuracy and mesh strategy of the FEM-*hp* method.

The impact of the error level for the FE-IIEE method, U, has not been taken into account so far. In the preceding experiments a value of $U < e_{hp}/10$, which assures that the truncation of the boundary itself has no effect on the accuracy of the *hp* method, was chosen. The impact of the ACA approximation on the FE-IIEE strategy will be outlined next with the help of an experiment similar to that of Fig. 9. The exact error of the field on S versus ϵ is depicted in Fig. 11 for several FE-IIEE iteration numbers. The *hp* mesh is deliberately chosen fine enough in this case to not limit the accuracy. It can be noticed that ACA only begins to have influence on the error when ϵ is greater that the actual error obtained. The floor of the error is determined in this case by the FE-IIEE method that, as expected, makes the floor gets lower as the number of iterations increases. The values of ϵ in the optimum range depend on and closely follow the value of U in percentage: the error obtained for $\epsilon = 0.1$ is very close to 0.1%, the error obtained for $\epsilon = 0.01$ is very close to 0.01%, and so on.

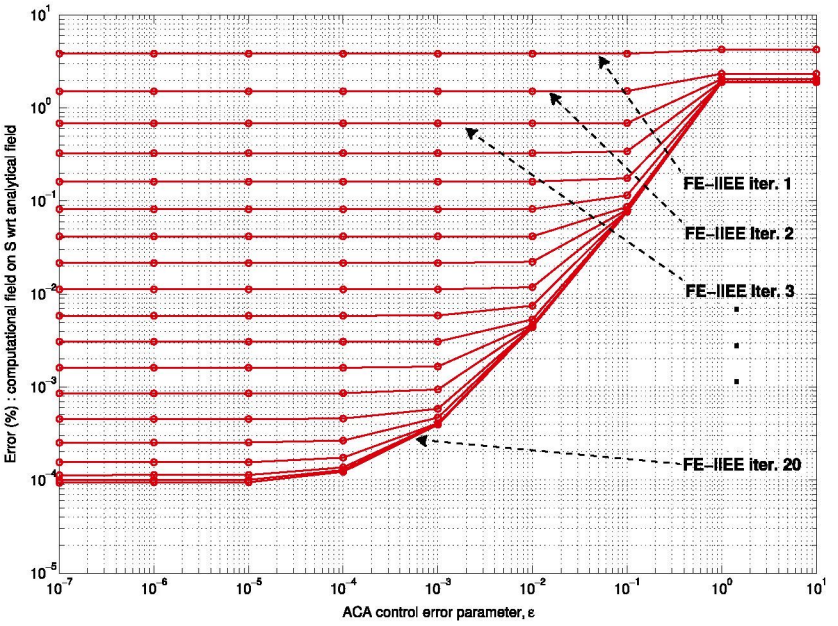

Fig. 11. Exact error of the field on S versus ϵ with the number of FE-IIEE iterations as parameter. Different curves correspond to different FE-IIEE iterations in a consecutive way, i.e., to different values of threshold U. A total of 20 FE-IIEE iterations are displayed. The curve corresponding to the first FE-IIEE-iteration appears at the top of the graphic. The curves corresponding to iterations 17th to 20th appear at the bottom (almost one on top of the other).

As the main conclusion from results above, it can be stated that a value for ϵ around $\epsilon = U = e_{hp}/10$ makes FE-IIEE with ACA transparent to the *hp* refinement strategy and does not interfere with the refinement strategy and accuracy of the self-adaptive *hp*-FEM, whereas an optimum level of ACA compression is gained.

5.2. Computational cost

Empirical results of the computer time expended in the FE-IIEE loop of the self-adaptive *hp*-FEM for a number of scattering simulations are presented next. Thus, empirical computational complexities are obtained. The so obtained computational complexities are compared with the estimates presented in Section 4; specifically, with the estimates corresponding to t_{ACA} and t_{INT} (expressions (33) and (30), respectively).

Sequences of mesh discretizations resembling both types of refinements, *h*-refinements and *p*-refinements, are considered. Also, two situations are contemplated for each type of refinement depending if the scatterer is of the same electrical size along the sequence, or if the electrical size changes. Thus, a total of four different types of "canonical" cases are considered:

- Constant electrical size of scatterer and constant polynomial order p.
  Several sequences of FEM meshes for the analysis of a given scatterer are considered in this experiment. Each sequence is generated by uniform *h*-refinements of a given initial mesh. The initial meshes are topologically the same but are distinguished by the use of a different order p. Thus, sequences of meshes corresponding to *h*-refinements and uniform p are created. Simulation runs with this sequence of meshes and identical frequency are performed. These sequences resemble the behavior of *h*-adaptive methods. Note also that *h*-refinements is what optimal *hp*-adaptivity locally delivers nearby a field singularity. Values of p ranging from $p = 2$ to $p = 8$ are considered. Thus, the performance behavior of FE-IIEE under *h*-convergence, also in the context of higher order methods, can be studied.
- Increasing electrical size of scatterer and constant polynomial order p.
  A sequence of simulation runs are performed in which the electrical size of a given scatterer is increased and the polynomial order p is kept constant.
  A sequence of meshes is generated for this purpose. The total number of dof is proportionally increased as the electrical dimensions get larger but maintaining constant the number of dof per wavelength over the boundaries. As polynomial order p is not modified, the electrical edge size h/λ of the finite elements on the boundaries remains constant within each sequence.
  As in the previous case, similar sequence of meshes with different values of constant p (ranging from $p = 2$ and $p = 8$) along the mesh are considered. The motivation for these experiments is to assess the improvement in performance and scalability achieved by ACA acceleration in *h*-convergence of FE-IIEE with the electrical size of the problem; also, in the context of higher order methods.

- Constant electrical size of scatterer and increasing polynomial order p.
  A sequence consisting of topologically identical meshes are generated corresponding to the analysis of a given scatterer but each mesh being associated to different polynomial order p. Simulations are performed using the same frequency. This experiment resembles the situation given by the use of p-adaptivity and spectral methods.
- Increasing electrical size of scatterer and increasing polynomial order p.
  A sequence of simulation runs are performed in which the electrical size of a given scatterer is increased as in the second case. However, the increment in the number of dof comes from the use of a higher p in each simulation run. It is worth noting that in this case the electrical edge size h/λ is modified from one mesh to the next one of the sequence in order to keep the number of degrees of freedom per wavelength constant. These experiments allow to assess the improvement in performance and scalability achieved by ACA acceleration in the p-convergence of FE-IIEE with the electrical size of the problem.

Each one of the four "canonical" cases is addressed in a subsection below.

The experiments mentioned above cover pretty much the type of situations given in an arbitrary hp-discretization. Thus, they somehow provide bounds about the performance and scalability to be expected on the use of ACA with higher order methods and hp discretizations in general, and specifically, when using FE-IIEE as mesh truncation technique for FEM analysis.

Computer times corresponding to the scattering of plane wave on a circular shape PEC cylinder are shown in the section. In all cases, computer times shown correspond to elapsed/wall time. Times shown are obtained after averaging the elapsed times corresponding to a few separate runs in the same dedicated computer. The computer used in the experiments is a EM64T server with 32 GB of RAM using an Intel E5620 processor at 2.4 GHz and with 12 MB of cache. It is worth noting that although the processor is quad-core, only one core is deliberately used in the experiments (only one thread is activated).

The circular shape has been chosen because its solution is somehow the paradigm of scattering smooth solutions. Scatterers giving field solutions with singularities, e.g., due to the existence of corners in the scatterer, have been deliberately avoided in the numerical experiments in order for the regularity of the solution does not interfere in the results. The availability of the analytical solution in the case of the circular shape scatterer is also handy to perform *a posteriori* checks of the exact error in each one of the cases and situations.

The FE-IIEE and ACA error thresholds, U and ϵ respectively, have been adapted to the exact error delivered by FEM without acceleration for each particular mesh, according to what has been referred to previously in Section 5.1 as "optimum" thresholds. Thus, neither ACA or FE-IIEE significantly influence the inherent precision of FEM-hp.

The clustering preprocess over basis functions associated to S and S' surfaces is based on a FFM type partitioning scheme. The number of ACA groups G has been chosen specifically to be $G_f = \sqrt{40\,P/\lambda}$, where P stands for the size of the perimeter of the minimum square enclosing the problem domain Ω^{FEM}. According to this clustering approach, the interaction corresponding to two touching groups is considered to be an electrically near interaction, and, thus, is computed via conventional integration. When the electromagnetic interaction corresponds to well-separated groups (separated by at least by one ACA group) it is considered to be a far-off interaction and hence is calculated by means of ACA. For the sake of fairness in the comparisons, N and N_{far} have been differentiated, being N_{far} the average number of unknowns considered by ACA.

5.2.1. Constant electrical size of scatterer and constant polynomial order p

For these experiments, sequences of meshes corresponding to the analysis of the scattering of a plane wave on a circular PEC cylinder of electrical size fixed to $R/\lambda = 5$ (where R is the radius) are considered. In all cases, the distance between the PEC and the boundary S', and between S and S' is 0.5λ. Note that FE-IIEE can work with much smaller distances between S and S' and with boundary S' on top of the PEC scatterer. Following the same reasons already explained in Section 5.1, we have selected the distances above mentioned.

A mesh sequence is obtained by reducing the discretization size h on S and S'. Thus, the number of unknowns N is increased, and consequently the accuracy is enhanced. Different sequences corresponding to different polynomial orders p have been considered; specifically, results corresponding to polynomial orders, $p = 2$, $p = 4$ and $p = 8$, are shown. Different initial sizes h/λ for the initial mesh are set for each sequence. Specifically, the range for the discretization size h on S and S' goes from 0.196 to 0.016 for $p = 2$, from 0.393 to 0.023 for $p = 4$, and from 0.393 to 0.057 for $p = 8$.

Computer times t_{ACA} and t_{INT} obtained with each sequence are shown in Fig. 12. The experimental data show that the complexity exhibited in this case by the integral without acceleration, which is above $O(N^2)$, is drastically reduced by ACA. Actually, complexities obtained with ACA are very close to linear $O(N)$ for low and medium orders and bounded by $O(N^{1.5})$ for very high order ($p = 8$ in the figure). Actually, the case of $p = 8$ is somehow extreme as it will be explained later. It is also clear that the computation time itself is reduced from one to almost two orders of magnitude due to the speed-up achieved by ACA. Moreover, the compression rates raise as the matrix size (or N) is incremented, achieving compression levels up to 98% when $p = 2$ and 95% when $p = 8$ for the upper limit of N.

The excellent performance of ACA in this case compared to conventional integration can be explained by the variation rate of the parameters involved in the expressions given in Section 4. As it is explained below, the key is the good behavior of the rank, r. As polynomial order p remains constant within each sequence its effect in the computational complexity arising from formulas (33) and (30) can be discarded. Thus, we easily conclude that computational complexity corresponding to t_{INT}

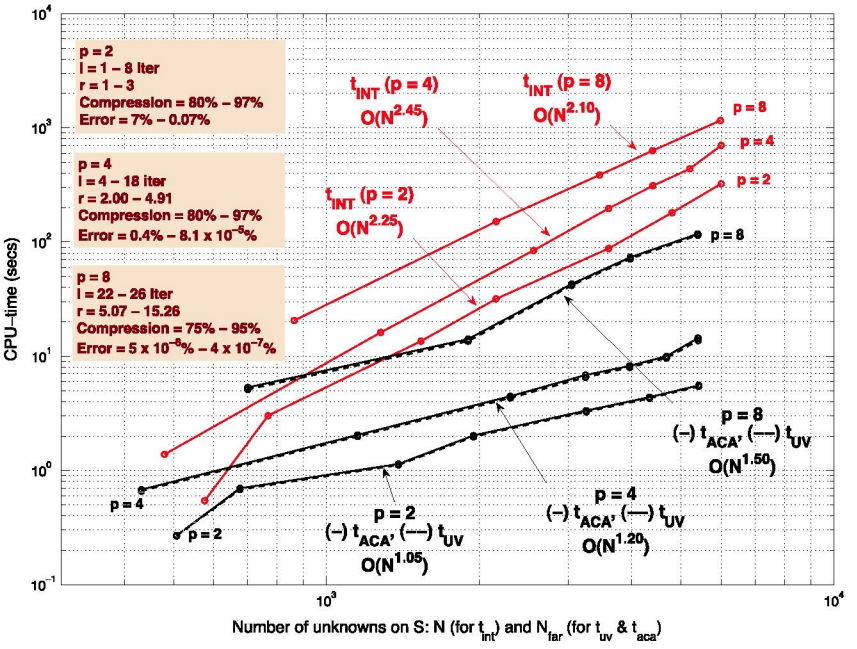

Fig. 12. Constant electrical size of scatterer and constant polynomial order p ($p = 2, p = 4, p = 8$).

is approximately $O(IN^2)$. In case of ACA acceleration, dependency on I can be neglected as $(I\, t_{\text{mult}})$ is negligible in comparison to t_{UV}; in other words, $t_{\text{ACA}} \approx t_{\text{UV}}$ (as it is observed in the figure where t_{ACA} and t_{UV} lines are practically one on top of the other). Thus, complexity in the case of using ACA acceleration depends on $(G_{\text{f}}\, rN)$ and $(G_{\text{f}}\, r^2 N)$ terms. As the clusterization strategy used with ACA only depends on the electrical size of the problem (scatterer), G_{f} remains constant in this experiment and can also be discarded in terms of complexity estimates. On the other hand, usually $r^2(2p+1)$ happens to be smaller than $r(12p^3 + 51p^2 + 69p + 30)$ since value of r is very low in practice. Taking all above into account, t_{ACA} is of order $O(rN)$.

Thus, it is clear the effect of ACA reducing the quadratic order dependency of the non-accelerated approach to a dependency of order $O(rN)$. Note that the minimum complexity achievable with ACA is linear (when variation of the rank r within the sequence is negligible) in contrast to straight integration that is at least quadratic. As the meshes of the sequence are obtained by h-refinement and the frequency is fixed (i.e., electrical size of the scatterer remains constant), the rank r remains pretty constant and, hence, t_{ACA} is close to linear as it can be observed in the figure. This is still true when we go high in p. The slight increment in the slope when going higher in p (e.g., $O(N^{1.2})$ for $p = 4$) is due to the faster decrement of the solution error when performing h-refinements on higher order meshes with respect to lower order meshes. For instance, as it is displayed in the legend of the figure, the error for $p = 2$ goes down two orders of magnitude (from 7% to 0.07%) while with $p = 4$ the error range is much larger (from 0.4% to 8.1 10^{-5}%) considering a similar interval of number of unknowns. The pretty constant behavior of the rank r, despite the fast growth of the submatrix blocks with N, is responsible of the increment of the rates of compressions achieved with N. On the other hand, the relatively low values of r are responsible of the high rates of compressions achieved (see data displayed in the legend of the figure).

However, it is observed that for $p = 8$ there is an inflexion point in the line that changes from almost linear complexity to complexity $O(N^{1.5})$ for larger N. This effect deserves an explanation. The separation of linear complexity behavior for $p = 8$ is not directly related to the use of very high p but with the very low error level that is achieved (in the order of 10^{-7}%) for the electrical size considered in this experiment. When this extreme accuracy is required, the sensibility of the rank with regard to ϵ is much higher being responsible of the separation of the complexity from linear; nevertheless, bounded by $O(N^{1.5})$ in the worst case. Although omitted in the paper the same experiment with $p = 8$ was run again with slightly different ϵ and the computational complexity obtained was significantly lower. In any case, computation time with ACA is one order of magnitude lower that when conventional integration approach is used.

5.2.2. Increasing electrical size of scatterer and constant polynomial order p

In this case, a sequence of simulation runs are performed in which the electrical size of a given scatterer is increased and the polynomial order p is kept constant. Specifically, the electrical radius of the circular scatterer, R/λ, varies from 1 to 25. The distance between the PEC boundary and S', and between S and S', are both equal to 0.25λ, i.e., it is not proportional to the electrical size of the scatterer. This is deliberately chosen in order to select a test scenario as similar as possible to what a final user would run. Therefore, a simple scaling in the frequency is not enough and different meshes have to be generated.

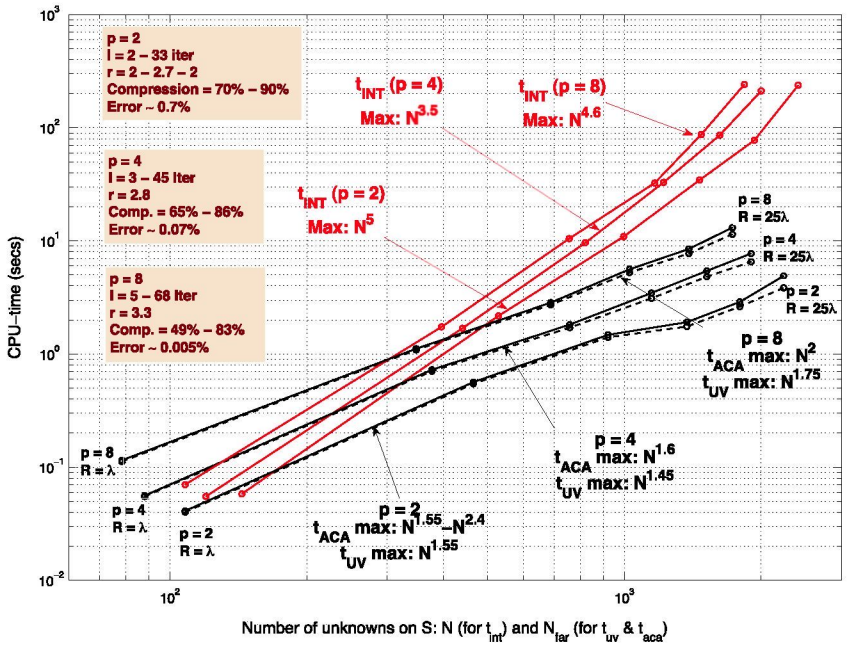

Fig. 13. Increasing electrical size of scatterer and constant polynomial order p ($p = 2, 4, 8$).

As in the previous case, several polynomial orders p have been considered; specifically, results corresponding to polynomial orders, $p = 2, 4, 8$, are shown. For a given p, the total number of unknowns is proportionally increased as the electrical dimensions grow, while maintaining constant the number of dof per wavelength, i.e., the electrical edge size h of the finite elements on the boundaries remains constant for a given p. But, depending on the selected p, different electrical sizes for the discretization are set for the meshes; specifically, $h/\lambda = 0.20$ for $p = 2$, $h/\lambda = 0.40$ for $p = 4$, and $h/\lambda = 0.80$ for $p = 8$ in order to keep constant the number of dof per wavelength.

Experimental results corresponding to computer times t_{ACA} and t_{INT} are displayed in Fig. 13. It is observed how computer times are drastically reduced by the use of ACA, and the computational complexity is drastically lowered. Only for very small electrical sizes and high polynomial orders p the time t_{INT} turns to be close or below t_{ACA}. Nevertheless, this is not of practical importance because the computational resources consumed with such electrical small problems are usually low, and the employment of fast methods is unnecessary in that scenario.

The behavior of t_{INT} and t_{ACA} can be explained in terms of the expressions given in Section 4. As in Section 5.2.1, it is easy to infer that computational complexity corresponding to t_{INT} is approximately $O(IN^2)$. The experimental results show scalability of order significantly higher than quadratic. This is due to the variation of the number of iterations I of FE-IIEE. For example, at the left parts of the plot where the electrical size of the scatterer is of a few wavelengths, the variation of I is about 4–6 depending on the order considered; but in the right part of the graphs, which covers electrical sizes from 20λ to 25λ, the increment in the number of iterations I is as high as 16 for order 2, 17 for order 4 and 20 for order 8. Thus, the high complexity observed in Fig. 13 for time t_{INT} is coherent with the variation of I according to the theoretical expectations. It is worth noting that the number of FE-IIEE iterations required for a given error can be reduced by incrementing the overlapping area between S and S'. However, for simplicity (and also illustration purposes) we have preferred to keep fixed the electrical distance between S and S' in the experiments.

With respect to computer time t_{ACA}, it is observed in the figure that $t_{\mathrm{ACA}} \approx t_{\mathrm{UV}}$, i.e., t_{UV} is the dominant term in ACA being $t_{\mathrm{mult}}\, I$ much smaller. The exception arises when the problem is electrically very large in which the difference between t_{ACA} and t_{UV} (although small) is noticeably in the figure. For instance, observe for $p = 8$ the change in the slope from 1.75 of t_{UV} to 2 for the total t_{ACA}. The reason is precisely the high number of iterations I of FE-IIEE (as high as 33, 45 or 68 for $p = 2, 4, 8$, respectively). On the other hand, and as it was explained before in Section 5.2.1, t_{ACA} is of order $O(G_{\mathrm{f}}\, rN)$ when considering p constant in expression (33). It is numerically observed that the rank is practically constant (see legend of the figure) with the clusterization strategy employed in which the number of groups is proportional to the electrical size of the perimeter of the problem; specifically, $G = \sqrt{40\, P/\lambda} \propto \sqrt{N}$. The rank depends on the field error requested but not of the problem size. Thus, we conclude that t_{ACA} should behave as $O(G_{\mathrm{f}} N) = O(N^{1.5})$. And that is what is approximately observed in the experiments. The deviation from slope equal to 1.5 for large scatterers is due to the increasing number of iterations I of FE-IIEE as the electrical problem size increases (noticeable only for large scatterers as mentioned above). The fact that in practice the theoretical dependency $G \propto \sqrt{N}$ is not always accomplished, specially with very high p ($p = 8$) in which large finite elements are employed, also influences the deviation from the theoretical slope of 1.5.

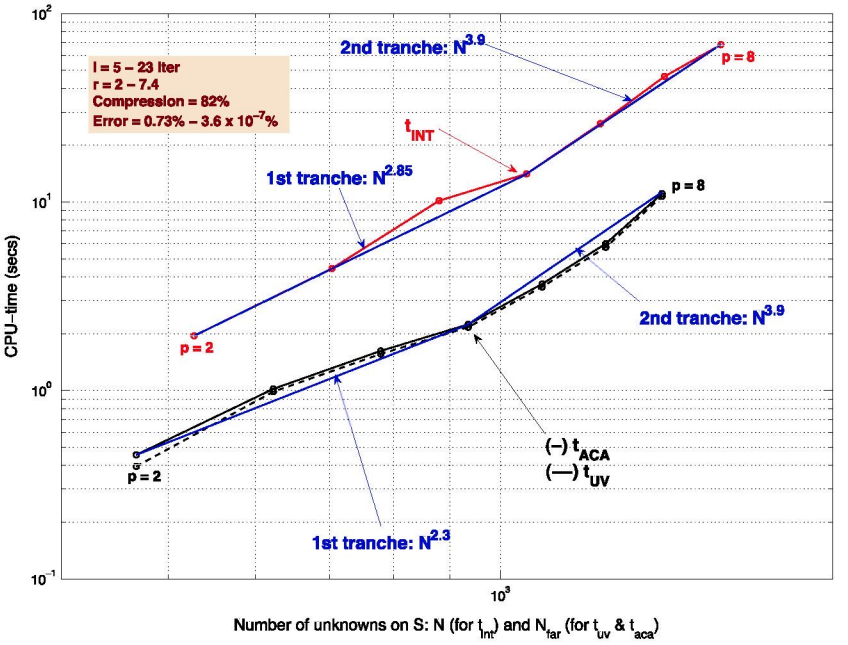

Fig. 14. Constant electrical size of scatterer and increasing polynomial order p.

5.2.3. Constant electrical size of scatterer and increasing polynomial order p

As in Section 5.2.1, analysis of the scattering of a PEC circular scatterer of electrical size fixed to $R/\lambda = 5$ (where R is the radius) is considered. The distance between the PEC and the boundary S', and between S and S' is 0.25λ. Topologically identical meshes are used for the analysis; they are simply associated to different polynomial order. Polynomial orders ranging from $p = 2$ to $p = 8$ with unit increments are considered.

For the first time in the experiments, variation with p is considered and, hence, conclusions about the performance of ACA in the context of p-adaptive methods and spectral methods can be obtained. Computer times corresponding to FE-IIEE using conventional integration, t_{INT}, and acceleration by ACA, t_{ACA}, are presented in Fig. 14. The experimental data show that the complexity exhibited in this case is similar for both approaches. As it is explained below, the reasons behind this behavior are very different depending if we consider t_{INT} of conventional integration or t_{ACA} corresponding to ACA acceleration. Nevertheless, the computation time t_{ACA} itself is roughly one order of magnitude lowered than t_{INT}.

By inspection of (30) we observe that the increase of computational complexity of t_{INT} with respect to the minimum, i.e., quadratic $O(N^2)$, is due to the number of FE-IIEE iterations I (as in previous cases) and also to variation of p itself. Although the number of iterations I does not increase as fast as in case of Section 5.2.2 where the electrical size of the problems changes, the increment of I from 5 for order 2, to 23 for order 8 along the curves, while the number of unknowns is only multiplied by 3 (from 500 to 1500), together with the increment of p itself, is enough to raise the computational cost to $N^{3.9}$. Note that computational cost is lower (close to $O(N^3)$) at the initial parts of the curve, though. It is important to emphasize the high increment in the accuracy (decrement of the error) from one step to the other of this experiment for a total range approximately of seven orders of magnitude (0.73% to $3.6\,10^{-7}$, see legend of figure). This is due to the use of increasing p on a problem (circular scatterer) with a smooth solution and hence, not limited by the regularity of the solution.

In the case of ACA, observation of (33) makes us to conclude that t_{UV} can be approximately reduced to $O(rNf\,(p^3, p^2, p))$, because the term with r^2, i.e., $O(r^2N\,(2p+4))$, can be neglected, specially for high orders. Note also that G_f has been obviated as it remains constant through the experiment because the electrical size of the problem is fixed. On the other hand, the term $O(rN\,(p+1))$ of (33) is much smaller than the terms of t_{UV} just mentioned. Also, and by observation of Fig. 14, we infer $t_{\text{ACA}} \approx t_{\text{UV}}$, i.e., $t_{\text{mult}} I \ll t_{\text{UV}}$.

The difference of the behavior of the computational complexity of ACA with respect to N in this case is two fold. On one hand, the variation of p itself explicitly affects t_{UV} through the leading term $O(rNf\,(p^3, p^2, p))$ of (33). On the other hand, the high variation of the error commented above due of the increment of p in the experiment makes the change in the rank to be significant and hence, affects t_{UV}. Both factors explain the deviation of computational complexity of ACA from its minimum $O(N)$. Nevertheless, in terms of computation time itself, ACA outperforms conventional integration for every order considered.

5.2.4. Increasing electrical size of scatterer and increasing polynomial order p

In this case, we combine the variation in p, as in the previous case of Section 5.2.3, with the variation of the electrical size of the problem as in Section 5.2.2. The electrical radius of the circular scatterer, R/λ, varies from 1 to 25. The distance

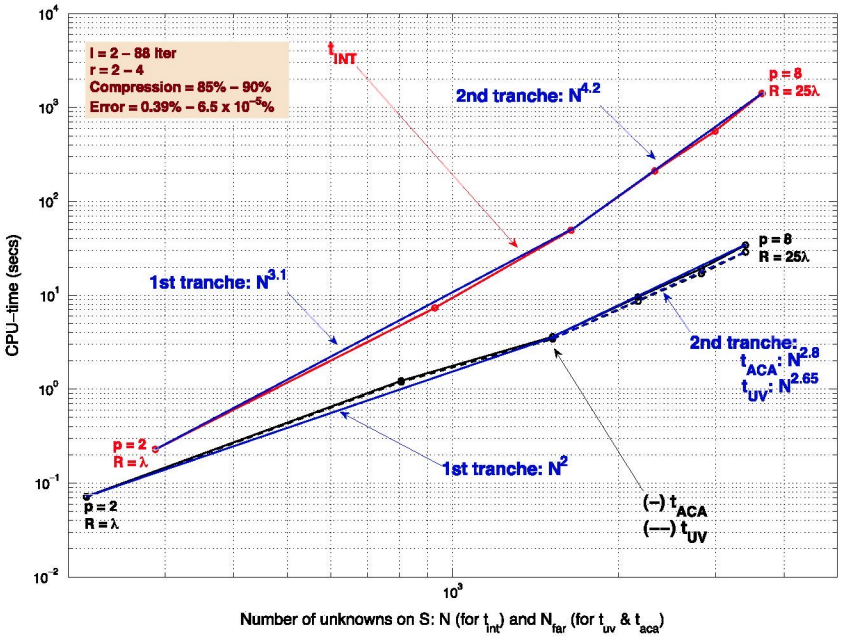

Fig. 15. Increasing electrical size of scatterer and increasing polynomial order p.

between the PEC boundary and S', and between S and S', are both equal to 0.25λ. Polynomial orders ranging from $p = 2$ to $p = 8$ with unit increments are considered. For the sake of a fair comparison, the meshes used in the experiment are generated in such a way that the number of unknowns per λ on S is kept constant; specifically, the number of unknowns per wavelength is set to a value of 20. As a consequence, the size h of the finite elements is slightly increased as p is raised.

Computer times t_{INT} and t_{ACA} are presented in Fig. 15. It is observed a significant reduction on computer time by the use of ACA acceleration; also the computational complexity of the algorithm FE-IIEE is lowered. Computational complexity with ACA is higher than in the case of increasing electrical size with fixed p and lower than in the previous case of constant electrical size with varying p. The explanations in terms of theoretical computer time expressions shown in Section 4 are given in the following.

With respect to t_{INT}, as explained in Section 5.2.2, the effect of the increase in the electrical size of the problem, specifically, the increase of the ratio between the scatterer area and the overlapping area (enclosed by S and S' boundaries), gives rise to a significant increase of I along the experiment. This results, according to (30), in an important increment in the computational complexity.

When analyzing the behavior of t_{ACA} displayed in Fig. 15 we observe $t_{\text{ACA}} \approx t_{\text{UV}}$, i.e., $t_{\text{mult}}\, I$ is much smaller than t_{UV}. The effect of $t_{\text{mult}}\, I$ is only noticeable when the problem is electrically very large. Specifically, the change in the slope from 2.65 of t_{UV} to 2.8 for the total t_{ACA} for large scatterers is due to the effect of t_{mult}. With respect to t_{UV} itself, we conclude from (33) applied to this case, that the computational complexity is approximately reduced to $O(G_f rNf\,(p^3, p^2, p))$ in which, as in the other cases, the term with r^2, i.e., $O(G_f\, r^2 N(2p+4))$ is smaller and can be neglected. The clusterization strategy leads to the number of ACA groups to be proportional to the square root of the perimeter of the scatterer, and hence, $O(G_f\, rNf\,(p^3, p^2, p))$ becomes $O(rN^{1.5} f\,(p^3, p^2, p))$ as in the case of increasing electrical size of scatterer of Section 5.2.2. The main difference is that in the present case the polynomial order p is no longer constant. In the present case of a circular scatterer, the solution is smooth and higher p makes the error decreases significantly with the number of unknowns despite the number of unknowns per wavelength remains constant. As we observed in the case of increasing p of Section 5.2.3, the rank mainly depends on the variation of the error. Thus, the rank will not be constant. However, the rank varies only from 2 to 4 in this experiment due to the increase of the electrical size that makes the error decreases not so drastically. Thus, the deviation of the computational complexity with ACA from the minimum of $O(N^{1.5})$ is moderate compared with the case of Section 5.2.3; specifically, around $O(N^{2.5})$ as shown in the figure. Nevertheless, note that the computational complexity of ACA is significantly lower than the one corresponding to conventional integration.

6. Conclusions

The implementation of the ACA algorithm in the context of a sophisticated self *hp*-adaptive software for open problems has been validated. The accuracy obtained is directly related with the error control parameter of ACA, ϵ, which can be easily tuned. It has been demonstrated that high levels of compression can be achieved without interfering in the accuracy or the refinement strategy of the *hp* adaptivity nor the boundary truncation method. The robustness of the ACA parameter has

been proved in this context. Although it has been shown that there is a trade-off in its election, there exists a margin for ϵ that allows to save a considerable amount of computational resources while maintaining accuracy. Only in the case of very high values of ϵ with respect to the requested *hp* error the refinement *hp* strategy may be misled for certain problems. A different value of ϵ for each *hp* iteration, within the order of $e_{hp}/10$, is recommended.

Performance of *h*-convergence and *p*-convergence strategies applied to the cases of a scatterer of fixed electrical size and of increasing electrical size has been explored. Theoretical estimates of the computational complexity of ACA and conventional integration in the context of FE-IIEE have been deducted predicting the empirical results of the performance tests made later. The minimum computational complexity predicted for conventional integration is $O(N^2)$. That quadratic complexity is reduced to a minimum linear complexity $O(N)$ for the case of fixed electrical size and to a minimum $O(N^{1.5})$ for the case of increasing the electrical size. Empirical computational complexities obtained in the tests deliver those minimum computational complexities when *h*-refinements are made. When using *p*-refinements computational complexities higher that the minimum were obtained. The main reason behind the deviation of the complexity in ACA with respect to its minimum is the behavior of the error that decreases very fast when *p*-refinements are made on a smooth solution problem such as the one selected for the tests. In other words, the influence of *p* in the computational complexity is not direct; but indirect through the good behavior of the error that affects the variation of rank in the ACA matrix blocks. Nevertheless, ACA notably reduces the computational complexity of the conventional integration in all cases with the exception of the specific test of *p*-convergence with a fixed electrical size smooth scatterer. Note this case is somehow paradigm of spectral properties of the convergence of the error with the use of *p*-refinements. However, in all cases, ACA outperforms conventional integration in terms of computational time; including the mentioned case of *p*-refinements over a fixed electrical size smooth scatterer.

The above results allow to conclude that ACA fulfills the requirements as fast method in the computation of the integral equation for a fully automatic adaptive software including *h*, *p* and *hp* versions. Also, it can be extrapolated the validity of ACA as fast method in the context of higher order methods in general.

Acknowledgment

This work has been supported by the Ministerio de Educacion y Ciencia, Spain, under Projects TEC2007-65214/TCM, TEC2010-18175/TCM, EC2013-47753-C3-2 and RTC 2014-23 80-4.

References

[1] L. Demkowicz, Computing with *hp* Finite Elements. I. One-and Two-Dimensional Elliptic and Maxwell Problems, Chapman & Hall/CRC Press, Taylor and Francis, 2007.
[2] L. Demkowicz, J. Kurtz, D. Pardo, M. Paszynski, W. Rachowicz, A. Zdunek, Computing with *hp* Finite Elements. II Frontiers: Three Dimensional Elliptic and Maxwell Problems with Applications, Chapman & Hall/CRC Press, Taylor and Francis, 2008.
[3] L.F. Demkowicz, Finite element methods for Maxwell equations, in: Encyclopedia of Computational Mechanics, John Wiley & Sons, Inc., 2004.
[4] L.E. García-Castillo, I. Gómez-Revuelto, F. Sáez de Adana, M. Salazar-Palma, A finite element method for the analysis of radiation and scattering of electromagnetic waves on complex environments, Comput. Methods Appl. Mech. Engrg. 194 (2–5) (2005) 637–655.
[5] I. Gómez-Revuelto, L.E. García-Castillo, M. Salazar-Palma, T.K. Sarkar, Fully coupled hybrid method FEM/high-frequency technique for the analysis of radiation and scattering problems, Microw. Opt. Technol. Lett. 47 (2) (2005) 104–107.
[6] R. Fernández-Recio, L.E. García-Castillo, I. Gómez-Revuelto, M. Salazar-Palma, Fully coupled hybrid FEM-UTD method using NURBS for the analysis of radiation problems, IEEE Trans. Antennas and Propagation 56 (3) (2008) 774–783.
[7] I. Gómez-Revuelto, L.E. García-Castillo, D. Pardo, L.F. Demkowicz, A two-dimensional self-adaptive *hp* finite element method for the analysis of open region problems in electromagnetics, IEEE Trans. Magn. 43 (4) (2007) 1337–1340. http://dx.doi.org/10.1109/TMAG.2007.892413.
[8] I. Gómez-Revuelto, L.E. García-Castillo, L.F. Demkowicz, A comparison between PML, infinite elements and an iterative BEM as mesh truncation methods for *hp* self-adaptive procedures in electromagnetics, Prog. Electromagn. Res. (PIER) 126 (2012) 499–519.
[9] P. Bettess, Infinite elements, Internat. J. Numer. Methods Engrg. 11 (1977) 54–64.
[10] J.P. Berenger, A perfectly matched layer for the absorption of electromagnetic waves, J. Comput. Phys. 114 (1994) 185–200.
[11] R. Coifman, V. Rokhlin, S. Wandzura, The fast multipole method for the wave equation: A pedestrian prescription, IEEE Antennas Propag. Mag. 35 (3) (1993) 7–12.
[12] N.A. Gumerov, R. Duraiswami, Fast Multipole Methods for the Helmholtz Equation in Three Dimensions, in: Elsevier Series in Electromagnetism, ELSEVIER, 2004.
[13] J.R. Phillips, J.K. White, A precorrected-FFT method for electrostatic analysis of complicated 3-d structures, IEEE Trans. Comput.-Aided Des. Integr. Circuits 16 (10) (1997) 1059–1072.
[14] J.R. Phillips, Error and complexity analysis for a collocation-grid-projection plus precorrected-FFT algorithm for solving potential integral equations with Laplace or Helmholtz kernels, in: Copper Mountain Conference on Multigrid Methods, 1995.
[15] E. Bleszynski, M. Bleszynski, T. Jaroszewicz, AIM: Adaptive integral method for solving large-scale electromagnetic scattering and radiation problems, Radio Sci. 31 (5) (1996) 1225–1251.
[16] S.M. Seo, J.F. Lee, A fast IE-FFT algorithm for solving PEC scattering problems, IEEE Trans. Magn. 41 (5) (2005) 1476–1479.
[17] S.M. Seo, J.F. Lee, A single level low rank IE-QR algorithm for PEC scattering problems using EFIE formulation, IEEE Trans. Antennas and Propagation 52 (8) (2004) 2141–2146.
[18] M. Bebendorf, Approximation of boundary element matrices, Numer. Math. 86 (4) (2000) 565–589.
[19] M. Bebendorf, S. Rjasanow, Adaptive low-rank approximation of collocation matrices, Computing 70 (1) (2003) 1–24.
[20] S. Börm, L. Grasedyck, W. Hackbusch, An Introduction to Hierarchical Matrices, Tech. Rep., Max-Planck-Institut fur Mathematik in den Naturwissenschaften, 2001, preprint No. 105.
[21] S. Börm, W. Hackbusch, Approximation of Boundary Element Operators by Adaptive $\mathcal{H}^2$-Matrices, Tech. Rep., Max-Planck-Institut fur Mathematik in den Naturwissenschaften, 2003.
[22] C.C. Lu, W.C. Chew, A multilevel algorithm for solving a boundary integral equation of wave scattering, Microw. Opt. Technol. Lett. 7 (10) (1994) 456–461.

[23] J.M. Song, W.C. Chew, Multilevel fast multipole algorithm for solving combined field integral equation of electromagnetic scattering, Microw. Opt. Technol. Lett. 10 (1) (1995) 14–19.
[24] J.M. Song, C.C. Lu, W.C. Chew, S.W. Lee, Fast illinois solver code (FISC), IEEE Antennas Propag. Mag. 40 (3) (1998) 27–34.
[25] I. Gomez-Revuelto, L.E. Garcia-Castillo, S. Llorente-Romano, D. Pardo, A three-dimensional self-adaptive *hp* finite element method for the characterization of waveguide discontinuities, Comput. Methods Appl. Mech. Engrg. (249–252) (2012) 62–74.
[26] I. Gomez-Revuelto, L.E. Garcia-Castillo, D. Pardo, High-accuracy adaptive modeling of the energy distribution of a meniscus-shaped cell culture in a petri dish, J. Comput. Sci. 9 (2015) 143–149. http://dx.doi.org/10.1016/j.jocs.2015.04.027.
[27] A. Quarteroni, A. Valli, Domain Decomposition Methods for Partial Differential Equations, Oxford Science Publications, 1999.
[28] L. Vardapetyan, L.F. Demkowicz, Full-wave analysis of dielectric waveguides at a given frequency, Math. Comp. 75 (2002) 105–129.
[29] C.A. Balanis, Advanced Engineering Electromagnetics, John Wiley & Sons, Inc., 1989.
[30] L.F. Demkowicz, A. Buffa, H^1, H(curl) and H(div)-conforming projection-based interpolation in three dimensions. Quasi optimal p-interpolation estimates, Comput. Methods Appl. Mech. Engrg. 195 (24) (2006) 4816–4842. See also ICES Report 04-22.
[31] I. Gómez-Revuelto, L.E. García-Castillo, M. Salazar-Palma, Goal-oriented self-adaptive *hp*-strategies for scattering and radiation problems, Prog. Electromagn. Res. (PIER) 125 (2012) 459–482.
[32] A.F. Peterson, S.L. Ray, R. Mittra, Computational Methods for Electromagnetics, IEEE Press, 1998.
[33] J.M. Jin, The Finite Element Method in Electromagnetics, second ed., John Wiley & Sons, Inc., 2002.
[34] R.F. Harrington, Time Harmonic Electromagnetic Fields, McGraw-Hill, Inc., 1961.
[35] R.F. Harrington, Field Computation by Moment Methods, IEEE Press, 1993.